

Пояснювальна записка
до дипломного проекту

на тему: Система інтерактивного пошуку інформації з використанням
поточного місцеположення користувача

Київ – 2019 року

ЗМІСТ

1.	Вступ.....	4
2.	Аналіз предметної області.....	6
2.1	Аналіз існуючих рішень:	7
2.1.1	Системи обробки і пошуку з використання місцеположення користувача:.....	8
2.1.1.1	Forsquare:.....	8
2.1.1.2	Google Maps:	9
2.1.1.3	Facebook\Instagram:	10
2.1.2	Провайдери даних місцеположень:.....	12
2.1.2.1	Google Maps Platform:	13
2.1.2.2	Factual:	13
2.1.2.3	Here maps:	14
2.1.2.4	OpenStreetMap:	14
2.1.3	Месенджери	15
2.1.3.1	WhatsApp.....	16
2.1.3.2	Telegram.....	18
2.1.3.3	WeChat.....	20
2.1.4	Зведена інформація про рішення.....	21
2.1.5	Висновки до розділу	24
3.	Структура системи	26
4.	Вимоги до системи	29
4.1	Налаштування Linux у розподіленому оточенні.....	29

					IA351.050БАК.001.ПЗ			
Зм.	Арк.	№ докум.	Підп.	Дата				
Розроб.		Клековкін Ю.М.			Система інтерактивного пошуку інформації з використанням поточного місцеположення користувача Пояснювальна записка	Літ.	Аркуш	Аркушів
Перевірів		Шинкевич М.К.				Т	2	60
Реценз.						КПІ ім. Ігоря Сікорського Каф. АУТС Гр. ІА-351		
Н. контр.								
Затв.								

4.1.1 VirtualEnv	32
4.1.2 Dependencies	33
4.2 Google Maps Platform як постачальних даних.....	37
4.2.1 Google Places API Request	39
4.2.2 Google Places API Reply.....	46
4.2.3 Google Maps URLs	48
4.3 Bot API та Telegram Messenger як інтерактивне оточення	50
4.3.1 Створення Telegram Bot	51
4.3.2 Запит координат	52
4.3.3 Надсилання повідомлення	53
5.Посібник користувача.....	56
6. Висновки	58
7. Список використаних джерел	59

1. ВСТУП

Питання пошуку інформації стоїть перед людством з найдавніших часів. Колись це були казки та оповідання, тож потрібно було знайти якогось оповідача щоб дізнатись нові відомості. З появою писемності стало легше, але потрібно було вміти читати да отримати доступ до дорогоцінних манускриптів або книг. Так людство поступово розвивалося до сьогодення.

Починаючи з популярного протоколу пошуку Gopher, за допомогою якого можна було знайти потрібні документи у TCP/IP мережах, до сьогоднішнього дня, коли користувачі можуть запитувати голосових асистентів таких як Siri від компанії Apple, Alexa від Amazon та Assistant від Alphabet, людство продовжує вдосконалювати системи пошуку.

Щодо пошуку інформації з використанням поточного місцезнаходження користувача людство до появи інтернету мало величезні складності. У військових цілях використовували людей, плітки, різного роду літаючі засоби. У цивільному людям потрібна була медична допомога, освіта, їжа та розваги. В основному інформацію шукали серед людей, тож було дуже важливо швидко знайти на місцевості людину що добре знається на тому що коїться. Або швидко пересуватись у просторі, щоб встигати знаходити відповіді.

З появою телеграфу, радіо, телефонного зв'язку інтернету, або, іншими словам швидкісного зв'язку – почали будувати системи пошуку що можуть оперативно інформувати про зміни у навколишньому світі, або місцезнаходження різних об'єктів у незнайомій місцевості. Зі збільшенням міст ця проблема стала гостріша.

Згідно з дослідженням від компанії Allied Market Research [1] ринок сервісів що використовує поточне місцезнаходження користувача динамічно зростає. Очікуваний складний річний темп приросту (CAGR) складає 26.6% з 2016 по 2022 роки та сягне \$61,897 мільйонів до 2022 року.

					IA351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		4

Лідером серед користувачів сервісів буде північно-американський континент. З точки зору дохідності лідерами будуть картографічні сервіси та навігаційні сервіси.

В той же час найбільший зріст очікується у області що пов'язана з соціальними мережами.

Тож на сьогоднішній день соціальні сервіси що базуються на використанні поточного місцеположення користувача є дуже актуальними.

Приклади такого використання можуть бути різноманітні – від реклами товарів які можуть бути потрібні користувачу у цей момент, до медичної допомоги або військового призначення.

Необхідно розробити інтерактивну неупереджену систему пошуку інформації яка буде використовувати поточне місцеположення користувача. Поточне місцеположення користувача може бути надано як GPS координатами так і за допомогою LBS. Результати пошуку повинні містити дані з місцеположення об'єкту та мінімальну інформацію. Дані користувача повинні залишатися приватними.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		5

2. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

У світі існує багато різноманітних систем пошуку інформації. З давніх часів гостро стоїть проблема взаємодії системи: пошукові комплекси \Leftrightarrow користувач \Leftrightarrow інтерфейси з введення даних, та доставки знайдених даних до користувача.

У сучасному світі, з розвитком технологій, це питання не втратило гостроти. Пошукові системи наразі мають свою різну специфіку по темах. У той самий час багатofункціональність пошукових систем є загальноприйнятим вектором розвитку. Для взаємодії з іншими системами використовується система викликів (API).

На сьогоднішній день набули великої популярності так звані “пошукові боти” які використовують сучасні та зручні месенджери як функцію інтерфейсу ведення та доставки даних до пошукових систем та у зворотному напрямку.

Месенджери являють собою сучасний та дуже персоналізований засіб комунікації людини. З початком використання електронних повідомлень, світ почав усвідомлювати що телефонні виклики та відповідь на них потребує від людини відволікання та великої концентрації на розмову. В той час коли за допомогою електронних повідомлень, людина може відповісти у зручній, можливо більш розгорнутій, формі, не відволікаючись від виробничого процесу, у безпеці, і, що дуже важливо, у зручній для цього час. Упродовж еволюції інтерактивних систем використовувалось багато засобів доставки повідомлень. У цій роботі буде проведено порівняльний аналіз сучасних месенджерів, таких як WhatsApp, Telegram, та WeChat.

Завдяки екосистемі у якій існують месенджери, а це, зазвичай, смартфони, дані, що користувач отримує від пошуковою системи, не потребують додаткової обробки та можуть бути використані миттєво. Це дає можливість використовувати інтерактивні системи пошуку інформації

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		6

індивідуально, та у будь який момент часу, незважаючи на географічне розташування.

Ця робота зосереджена на проекті пошуку інформації з використанням місцеположення користувача з інтерфейсною частиною у вигляді месенджера.

Необхідними вимогами до такої системи є:

- 1) Безпека;
- 2) зручність для користувача у наданні точного запиту, який містить у собі географічні координати;
- 3) неупередженість - відсутність впливу на результати пошуку третіх сторін;
- 4) висока точність відповідей на питання користувача;
- 5) розвинуту та актуальну базу знань;
- 6) швидкість;
- 7) зручність використання отриманих даних.

2.1 Аналіз існуючих рішень:

При аналізі існуючих рішень було розглянуто ряд систем, в яких одним з основних завдань є обробка і пошук з використанням місцеположення користувача, також досліджені основні провайдери даних місцеположень, та месенджери як модель інтерфейсу для спілкування з користувачем, серед них:

- 1) Системи обробки і пошуку з використанням місцеположення користувача: Forsquare, Google Maps, Facebook\Instagram;
- 2) Провайдери даних місцеположень: Google Maps Platform, Factual, OpenStreet Maps;
- 3) Месенджери: WhatsApp, Telegram, та WeChat.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		7

2.1.1 Системи обробки і пошуку з використання місцеположення користувача:

Пошук за місцеположенням користувача це система що складає собою пошуковий рушій, базу даних місцеположень, та інтерфейс для отримання даних від користувача.

Для визначення місцеположення користувача використовують технологію GPS або ін. супутникові системи для визначення місцезнаходження. Заздалегідь відома інформація про розташування базових станцій мереж мобільного зв'язку GSM, UMTS та ін може також визначати місцеположення за допомогою триангуляції. Точність може коливатись від кілька кілометрів до 50 метрів, наприклад у LTE або WCDMA. Також можуть бути використані дані про доступні навколо точки доступу WiFi та їх розташування.

У цій роботі ми будемо використовувати на дані про місцезнаходження які надає інтерфейс користувача.

2.1.1.1 Forsquare:

Соціальна мережа збудована на отриманні та маніпуляціями з місцеположення користувача призначена для роботи на мобільних пристроях. Сервіс доступний не тільки користувачам з пристроями, що обладнані GPS-навігацією, наприклад користувачам смартфонів, але й просто для роботи з будь-яким мобільним телефоном. Якщо телефон не обладнаний GPS, то місцезнаходження визначається за допомогою сервісу LBS. Користувачі відмічаються («check-in») у різноманітних закладах за допомогою мобільної версії веб-сайта, SMS-повідомлення або ж спеціальної програми, розробленої під певну операційну систему мобільного телефону.

Згідно зі статистикою на 2017 рік, загальна кількість користувачів foursquare в Україні становить близько 60 000 користувачів у різних куточках країни.

Інтерактивний інтерфейс пошуку складає собою мобільний додаток доступний на всіх сучасних мобільних платформах. Або, як показано на

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		8

рисунку 2.1, інтерфейсу для браузерів. Система надає категорійний каталог місць з яких користувач може вибрати. Або питає що користувач може додати до бази даних об'єктів. Інформація у каталозі сортована за уподобанням інших користувачів. Кожен об'єкт обов'язково має фото, текстовий опис, рейтинг та адресу.

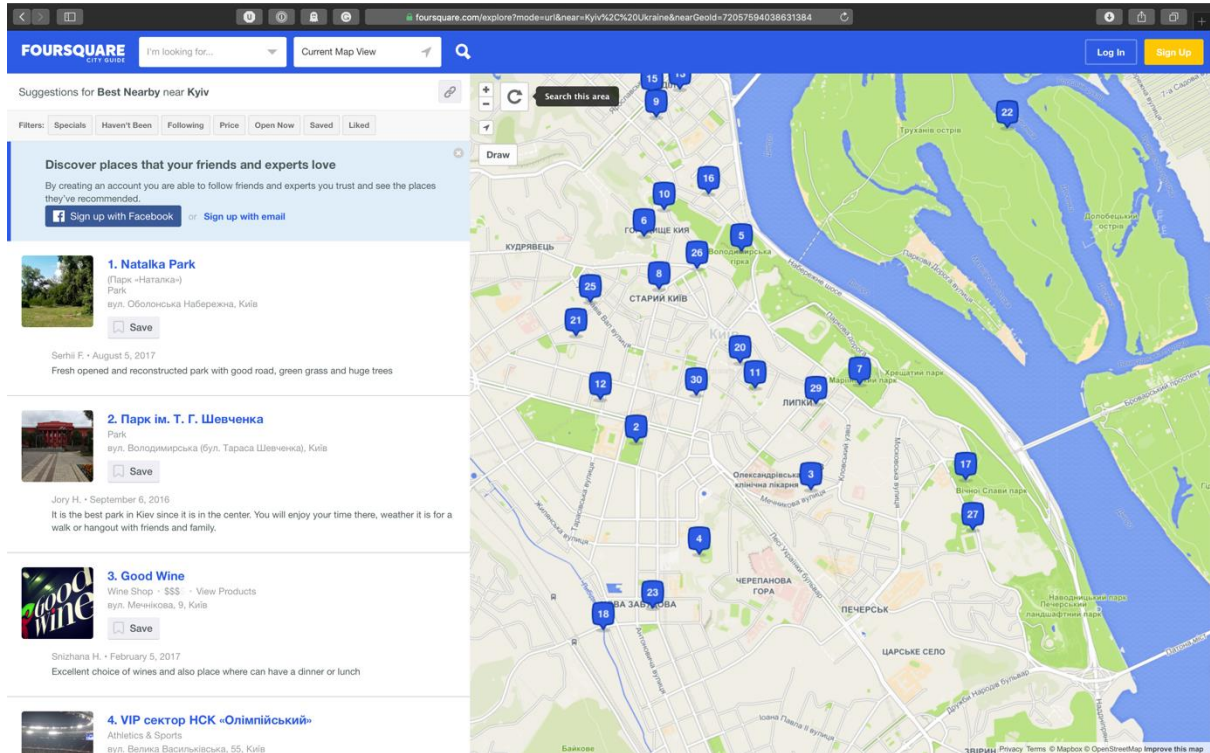


Рисунок 2.1 – Інтерфейс системи Forsquare

2.1.1.2 Google Maps:

Це сервіс мап для мобільних пристроїв та комп'ютерів. В обмін на дані про своє місцезнаходження та мовчазну згоду отримувати рекламу, користувачі мають змогу користуватись мапою не сплачуючи гроші.

Сервіс являє собою мапу та супутникові знімки земної кулі (а також Місяця і Марса) і надає користувачам можливості панорамного перегляду вулиць (Google Street View), аналізу трафіку у реальному часі (Google Traffic), прокладання маршруту (автомобілем, пішки, велосипедом або громадським транспортом). Сервіс інтегрований з каталогом-місць/об'єктів і картою автомобільних доріг, з пошуком маршрутів.

Інтерактивний інтерфейс, як показано на рисунку 2.2, складає собою веб-аплікацію, або окремий мобільний додаток у якому система

використовує поточне місцеположення користувача за допомогою GPS або LBS. Запити потрібно робити використовуючи природну мову. У відповідь на запит користувач отримує спеціальним чином сортований та підготовлений персоналізований набір даних у графічному представленні. Таким чином пошукова система впливає на користувача та підштовхує зробити вибір на користь раніше передплаченого контенту або звужує коло пошуку до місць у яких користувач був помічений раніше.

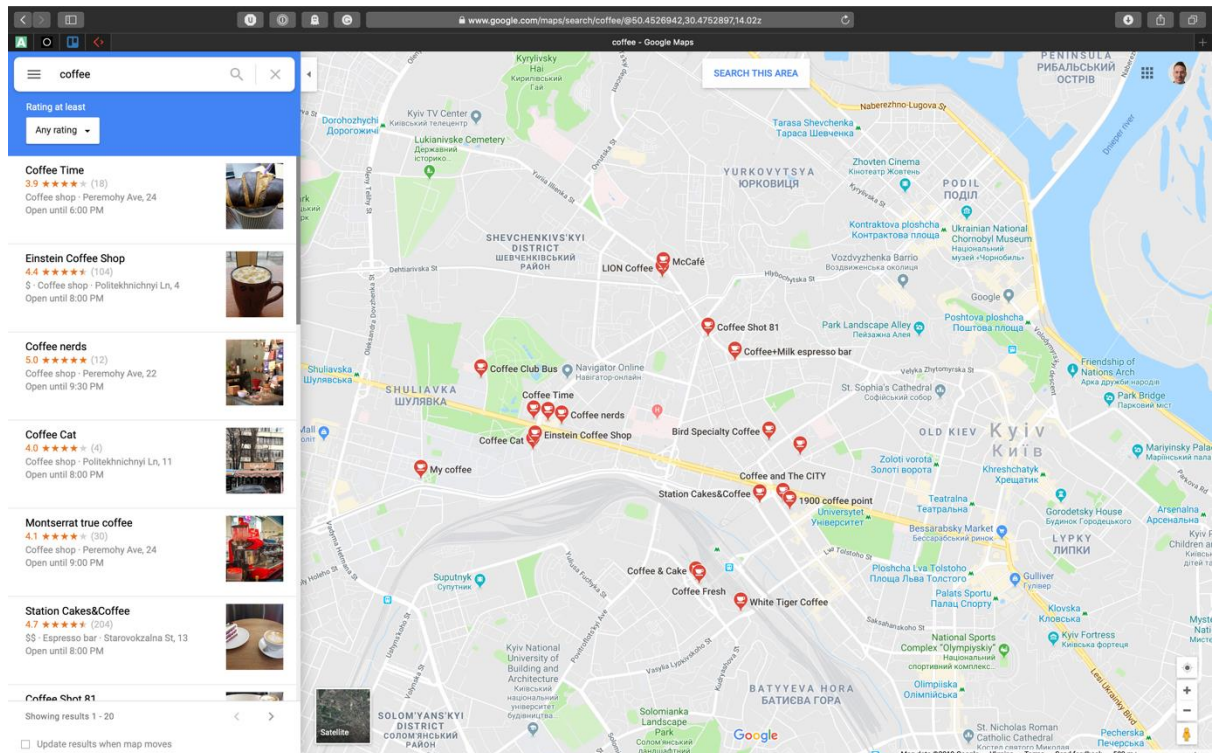


Рисунок 2.2 – Інтерфейс Google Maps

2.1.1.3 Facebook\Instagram:

Facebook - найбільша соціальна мережа світу. Почала працювати 4 лютого 2004 році, як мережа для студентів американського університету де навчався засновник та голова сервісу Марк Цукерберг. Станом на березень 2019 року кількість щомісячно активних користувачів була 2.3млрд, з яких 1,56 млрд відвідує свої профілі кожного дня. В ці дані включені всі користувачі що користуються сервісами компанії Facebook, тобто: Facebook, Instagram, WhatsApp або Facebook Messenger. Станом на січень 2019,

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		10

кількість користувачів що використовують застосунки компанії на мобільних приладах складає 96%. [2]

Instagram - соціальна мережа, що базується на обміні фотографіями, дозволяє користувачам робити фотографії, застосовувати до них фільтри, а також поширювати їх через свій. За появи цього сервісу дуже популярним став жанр айфонографії. На той час камери у мобільних телефонах компанії Apple були найпотужнішими, а застосунок дуже зручним в користуванні. Для Instagram найбільш підходять фотографії в квадратній формі - такого ж розміру як були у Kodak Instamatic і Polaroid. Зі співвідношенням співвідношення сторін 3:2. Ставши найпопулярнішим застосунком компанія була куплена Facebook за 1млрд доларів США. Що створило також ще один термін для оцінки капіталізації компаній - 1 інстаграм.

Наразі обидві соціальні мережі належать компанії Facebook, вони використовують одну й туж саму систему отримання та пошуку даних. Високої популярності досягли також функції пошуку з використанням поточного місцеположення.

Інтерактивний інтерфейс - веб аплікація або мобільний додаток. Приклад інтерфейсу показано на рисунку 2.3. Для пошуку за поточним місцеположенням користувача частіше використовують мобільний додаток. У моделі пошуку цієї системи первісним є пошук на базі запиту зробленого природною мовою. І вже до цього запиту додається поточне місцеположення користувача з використанням GPS або LBS. У відповідь на пошуковий запит користувач отримує посилання на сторінки компаній або закладів що зареєстровані у соціальній мережі. Компанії мають можливість впливати на видачу за допомогою передплати активної реклами. Пошукові механізми враховують попередню активність користувача, його уподобання. Тож користувач отримує набір упереджених даних.

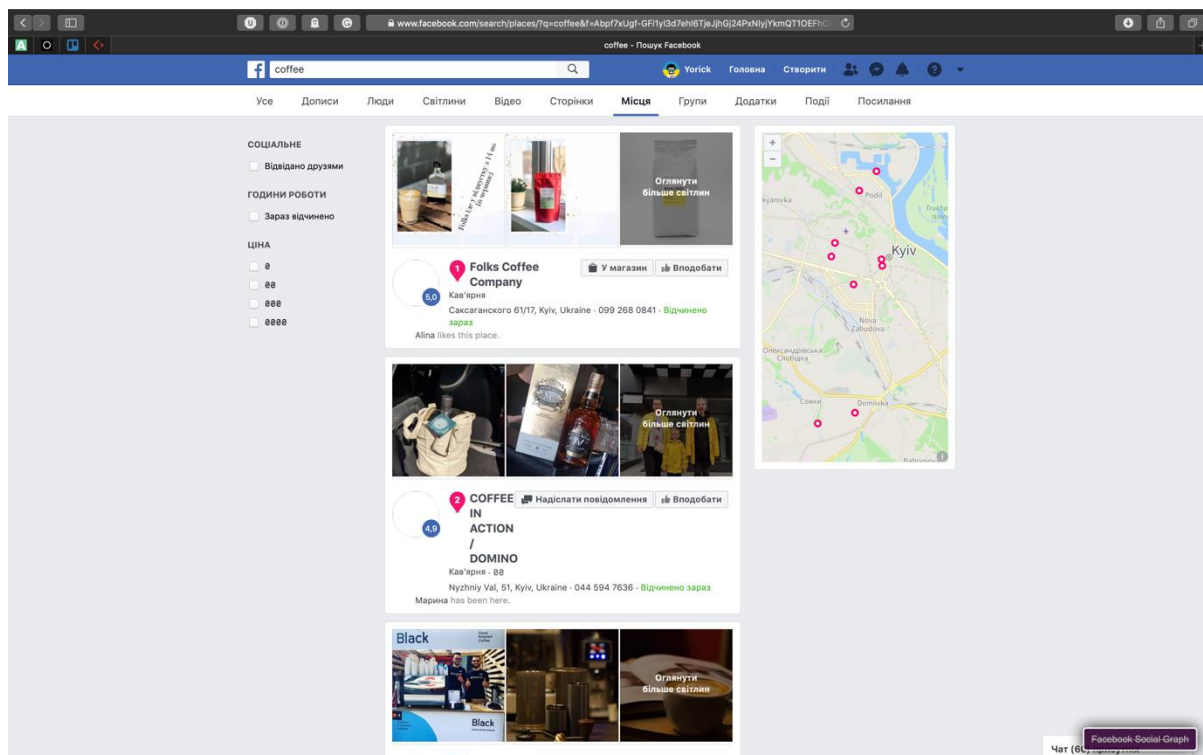


Рисунок 2.3 – веб-аплікація Facebook

2.1.2 Провайдери даних місцезнаходжень:

Людство здавна накопичує дані про місцезнаходження тих чи інших об'єктів. Такими об'єктами можуть бути цілі материки або континенти, гори, озера, океани, місця знаходження ресурсів, міста, окремі будівлі у містах. А у сучасному світі оцифровано мільярди різних об'єктів. Деякі компанії досі збирають дані від постачальників на місцях та актуалізують наявні дані у великі бази даних об'єктів. За допомогою цих даних людина має можливість дуже точно визначати та оперативно користуватись даними про світ.

Всі провайдери даних, для взаємодії використовують прикладний програмний інтерфейс (інтерфейс програмування застосунків, інтерфейс прикладного програмування) (англ. Application Programming Interface, API) - набір визначень підпрограм, протоколів взаємодії та засобів для створення програмного забезпечення. Спрощено - це набір чітко визначених методів для взаємодії різних компонентів. API надає розробнику засоби для швидкої розробки програмного забезпечення. API може бути для веб-базованих

систем, операційних систем, баз даних, апаратного забезпечення, програмних бібліотек.

2.1.2.1 Google Maps Platform:

Однією з найбільших агрегаторів географічних даних, даних об'єктів, на сьогоднішній день є Google Maps.

Картографічний сервіс від компанії Google також надає багато різноманітних функцій для розробників та компаній яким потрібно здійснювати пошук для надання своїх послуг. Так за допомогою Google API, що є частиною Google Cloud кожен може надсилати запити до бази даних та отримувати розгорнуту відповідь.

Ці сервіси є безкоштовними для певної кількості запитів до різних частин Google API. Наприклад для Google Places, для функції пошуку якогось об'єкту за координатами та пошуковим словом, безкоштовно можна використовувати 11000 запитів на місяць упродовж року (дані травень 2019). Тож недивлячись на безкоштовність компанія залишає за собою право змінювати кількість запитів та умови надання доступу.

Для пошуку можна використовувати логічні вирази, ключові фрази, запити на природній мові.

2.1.2.2 Factual:

Це потужний постачальник даних заснованих на поточному місцеположенні користувача. Мають багато функцій таких як визначення наближення користувача до конкретного місця, вимірювання відстані від користувача та ймовірність його появи у конкретному місці, аналітика поведінки користувача на підставі його пересувань у просторі та часі, визначення місцеположень місць за запитом та географічних координат.

Компанія працює здебільшого з даними у Северній Америці та Східній Азії де є лідером та співпрацює з бізнесами включеними в список Fortune 50.

Безкоштовних сервісів компанія не надає. Можливо отримати дуже

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		13

обмежений доступ для академічних досліджень.

2.1.2.3 Here maps:

Раніше Nokia maps, картографічний сервіс, який належить групі компаній, в який входять компанії AUDI AG, BMW Group і Daimler AG; використовується в численних пакетах програмного забезпечення, включаючи системи навігації (Garmin, BMW, Nissan) доступний для веб-браузерів і телефонів на різних платформах. Мапи включають такі функції, як пошук по мапі, супутникові мапи, побудова маршрутів, 3D-мапи, відображення пробок в реальному часі.

Застосунки Here багато в чому ґрунтуються на технології фірми Navteq, яку Nokia купила у 2007-2008 роках.

Безкоштовний доступ дозволяє 250000 транзакцій для не більш ніж 4999 унікальних користувачів щомісяця.

Цей постачальник даних є лідером за кількістю функцій які можуть бути використані безпосередньо у програмних продуктах. Зокрема це включає у себе пошук конкретного місця по координатах та текстовому запиту, надання зображень мапи з місцеположенням, погодних умов, надання даних про трафік, опрацювання даних для цілих автопарків, візуалізацію географічних даних, дані для реклами.

Цей постачальник даних має найбільш налагоджений контакт з локальними постачальниками даних, що є важливою конкурентною перевагою.

2.1.2.4 OpenStreetMap:

Це відкритий проект спрямований на збір, збереження та розповсюдження загальнодоступних геопросторових даних, створення інструментів для роботи з ними силами спільноти волонтерів. Проект заснований у Великій Британії в липні 2004 року Стівом Костом. У квітні 2006 року зареєстровано фонд OpenStreetMap – міжнародна некомерційна

					IA351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		14

організація, створена для підтримки розвитку та розповсюдження геопросторових даних, а також надання можливості використання геопросторових даних будь-ким.

Геопросторові дані проекту OpenStreetMap ліцензуються на умовах Open Database License, яка дозволяє їх використання з будь-якою, у т.ч. комерційною метою, за умови зазначення походження даних. Документація, генеровані зображення, які візуалізують геопросторові дані на головному сайті проекту OpenStreetMap, поширюються на умовах ліцензії Creative Commons Attribution-ShareAlike 2.0 (CC-BY-SA).

На протипагу власницьким наборам даних, таким як Google Maps, ліцензія OpenStreetMap гарантує вільний доступ до усіх наявних даних.

API OpenStreetMap включає в себе можливості пошуку або додавання просторових зображень, їх редагування, пошуку анотацій.

Цей провайдер даних являє собою одного з найпопулярніших проектів спільноти відкритого та вільного програмного забезпечення. В той самий час неоднорідність даних, та спорадичність оновлень які залежать виключно завдяки діяльності спільноти роблять OpenStreetMap досить слабким конкурентом на ринку постачальників даних.

2.1.3 Месенджери

Миттєві повідомлення або, повніше, система обміну миттєвими повідомленнями (англ. Instant messaging, скорочено IM) --телекомунікаційна служба для обміну текстовими повідомленнями між комп'ютерами або іншими пристроями користувачів через комп'ютерні мережі (як правило через інтернет). Зазвичай це були невеликі текстові повідомлення. Але з розвитком у систему були додані функції передавання файлів, зображень, звукових сигналів та повідомлень, відео, а також здійснення спільних дій, таких як малювання або ігри.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		15

Для користування цим видом комунікації необхідна клієнтська програма. Клієнтську програму системи миттєвих повідомлень часто називають інтернет-пейджером або месенджером.

Відмінність миттєвих повідомлень від, наприклад, електронної пошти тут в тому, що обмін повідомленнями відбувається в реальному часі. При відправленні повідомлення по електронній пошті повідомлення зберігається у поштової скриньці на сервері. Для того, щоб отримати повідомлення, отримувач повинен сам або за допомогою спеціальних додаткових програм перевірити свою поштову скриньку і забрати або прочитати їх. У інтернет-месенджерах відправлене повідомлення одразу передається користувачу на всі пристрої що зараз онлайн або за допомогою PUSH запитів.

Обмін повідомленнями може бути або між двома, або між декількома співрозмовниками (конференція, чат).

Система миттєвих повідомлень працює за різними протоколами. У безсерверних протоколах (FChat, NASSI, UChat) повідомлення передаються безпосередньо від одного співрозмовника до іншого. У серверних популярним довгий час залишався XMPP.

Завдяки доступності на мобільних пристроях та на практично усіх існуючих платформах, у сучасному світі месенджери являють собою найбільш поширений інтерактивний інтерфейс у кишені кожної людини світу.

2.1.3.1 WhatsApp

Месенджер із закритим вихідним кодом для смартфонів. Дозволяє пересилати дописи, зображення, відео та аудіо. Клієнт працює на платформах усіх мобільних платформах та на персональних комп'ютерах за виключенням OS Linux. Інтерфейс застосунку для MacOS показано на рисунку 2.4.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		16

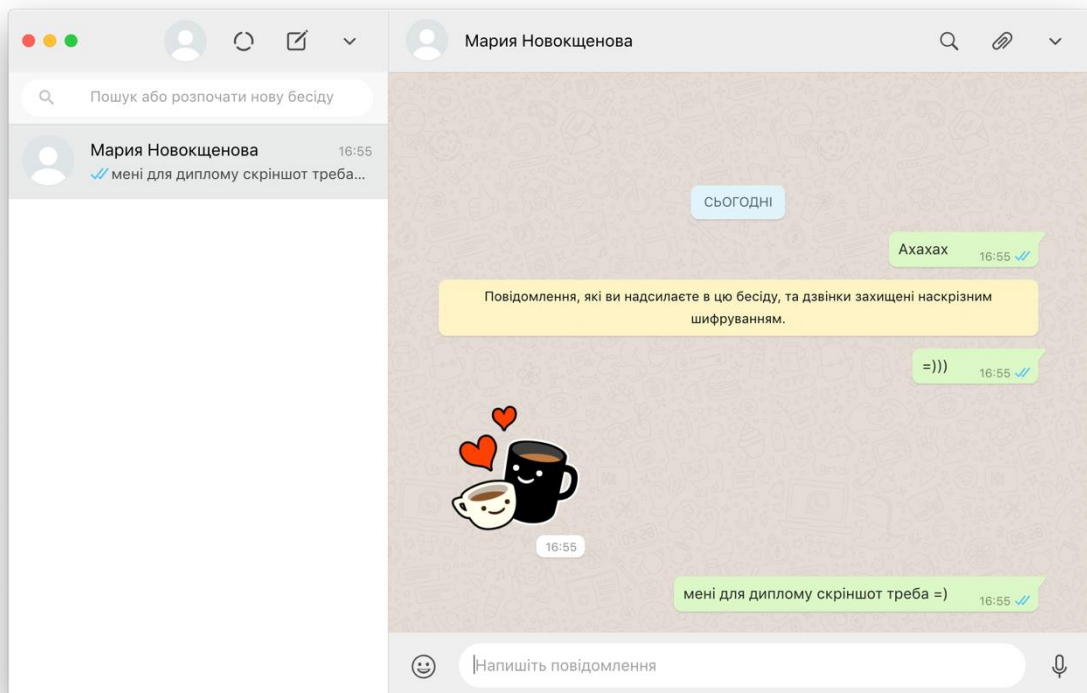


Рисунок 2.4 - Інтерфейс застосунку WhatsApp

Станом на грудень 2017 року цей месенджер має 1.5млрд щомісячно активних користувачів [2].

Як і всі продукти від компанії Facebook, меседжер має великі проблеми з безпекою. Проблеми сягають державного значення: наприклад розслідування Канади та Данії. Головним питанням, яке розглядалось у розслідуванні, було те, що програма завантажує телефонну книгу на сервери WhatsApp. Це робиться для того, щоб було зрозуміло, хто з контактів є користувачами месенджера, а хто ні. Це зручний спосіб пошуку серед контактів, а для швидкості роботи вся телефонна книга зберігається і оновлюється у реальному часі. Facebook наполягає що на серверах зберігається тільки список номерів телефонів без додаткової приватної інформації (імена, дати народження, додаткові відомості).

8 жовтня 2013 року сайт програми було зламано невідомими хакерами з Палестини. На сайті було розміщено інформацію про спірний статус палестинських земель.

У листопаді 2014 у застосунок додали функцію звітів про читання, яка надсилає користувачеві звіт про те, що надіслане повідомлення було відкрито отримувачем. Згодом вийшло оновлення, в якому цю функцію можна було відключити, щоб статус повідомлень ніколи не мінявся на «прочитано». А вже у лютому 2015 студент Датського університету Maikel Zweerink виклав у вільний доступ програму, яка дозволяла слідкувати за зміною статусів користувачів, бачити оновлення їхніх фото профілю, налаштувань безпеки, а також бачити статуси повідомлень незалежно від того, які налаштування вказав користувач.

У липні 2017 року експерти правозахисної некомерційної організації Electronic Frontier Foundation склали рейтинг компаній, який враховує рівень захисту користувачів від влади. WhatsApp була на першому місці зі списку з 26 американських компаній що найгірше зберігають приватність. Оцінка учасників проводилася на підставі низки критеріїв, серед яких прозорість взаємодії компаній з державою, заборона на передавання особистих даних третім особам, перевірка на легальність NSL-запитів і відстоювання змін у «секції 702» закону FISA, який дозволяє повсюдне стеження за користувачами як усередині країни, так і за її межами.

Незважаючи на великі проблеми з безпекою, цей месенджер залишається популярним на територіях Північної Америки та Західної Європи.

2.1.3.2 Telegram

Вільний месенджер з відкритим кодом що дозволяє обмінюватися текстовими повідомленнями та різноманітними файлами, зокрема графічними файлами та відеофайлами, сплачувати товари або послуги, а також безкоштовно телефонувати іншим користувачам програми. Інтерфейс месенджеру для системи MacOS показано на рисунку 2.5

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		18

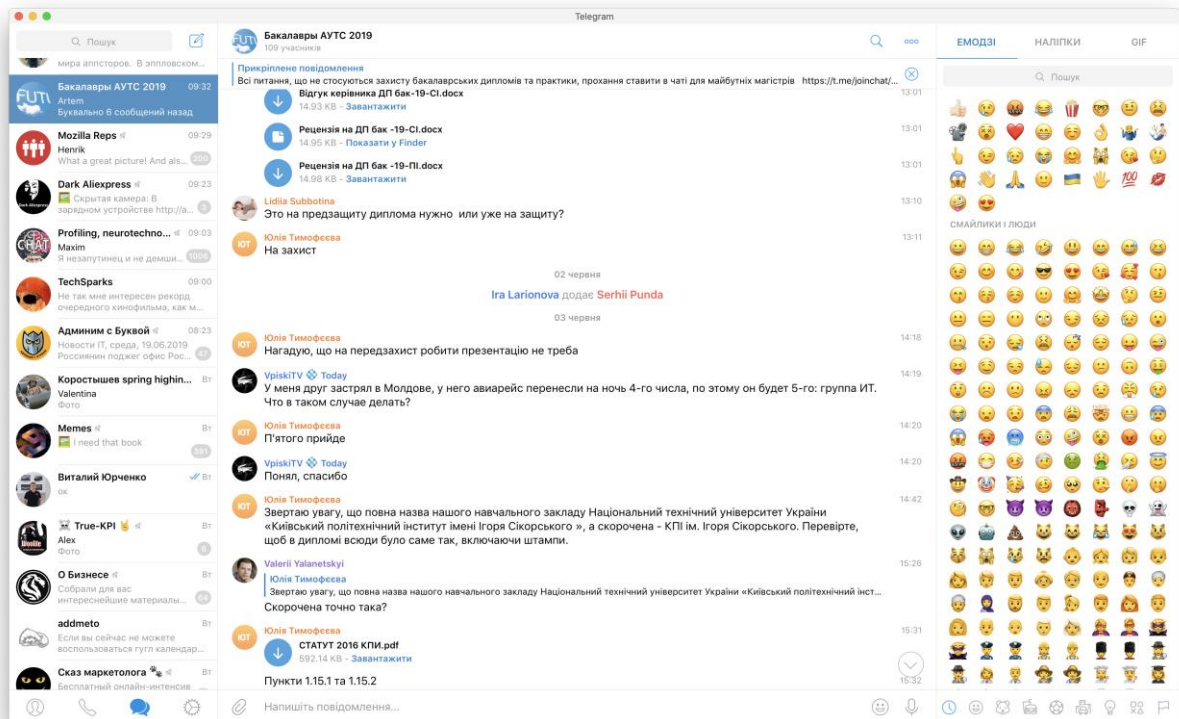


Рисунок 2.5 – Інтерфейс месенджеру Telegram

Для месенджера був створений протокол MTProto, що передбачає використання декількох протоколів шифрування. При авторизації і аутентифікації використовуються алгоритми RSA-2048, DH-2048 для шифрування, при передачі повідомлень протоколу в мережу вони шифруються AES з ключем, відомим клієнту і серверу. Також застосовуються криптографічні геш-алгоритми SHA-1 і MD5.

Безпека від перехоплення повідомлень, що пересилаються з боку сервера Telegram, забезпечується лише в режимі «секретних» чатів (Secret Chats), доступному з 8 жовтня 2013 року. Цей режим реалізує шифрування, при якому відправник і одержувач мають спільний лише для них ключ (end-to-end шифрування), із застосуванням алгоритму AES-256 у режимі IGE (англ. Infinite Garble Extension) для повідомлень, що пересилаються. На відміну від звичайного режиму, повідомлення в секретних чатах не розшифровуються сервером, історія листування зберігається лише на тих двох пристроях, на яких був створений чат.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		19

Станом на березень 2019 випадки зламу або проблем з безпекою не виявлено.

Один з найпоширеніших месенджерів на після-радянському просторі. Також великої популярності набув на територіях ісламістів.

Намагання деяких держав заборонити використання Telegram за допомогою фільтрації за айпі, або фільтрації трафіку з допомогою deep packet inspection не має ніякого результату – все працює дуже стабільно за швидко.

2.1.3.3 WeChat.

Мобільна платформа для обміну текстовими та голосовими повідомленнями, розроблена компанією Tencent. Станом на четвертий квартал 2018 року аудиторія WeChat склала 1096.7 млн. активних користувачів [2]. Інтерфейс месенджеру показано на рисунку 2.6

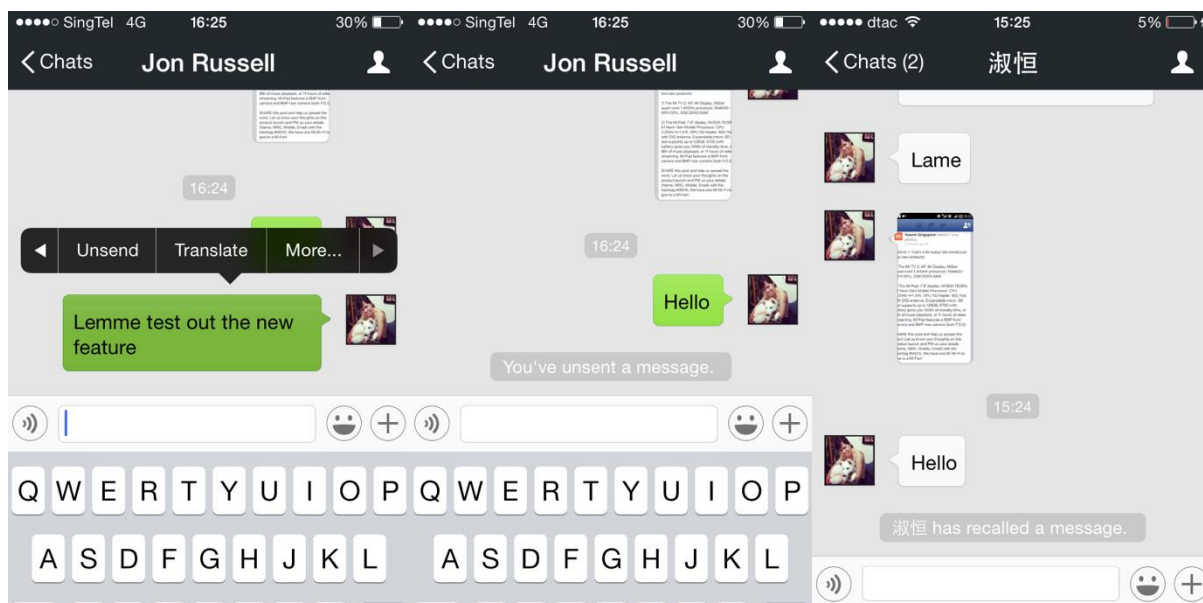


Рисунок 2.6 – Інтерфейс месенджеру WeChat.

В середині додатку є можливість здійснювати грошові перекази та робити оплату послуг. В Китайський новий рік 2016 через додаток було проведено більше транзакцій, чим у сервісу PayPal за увесь 2015 рік Пошук китайською фільтрується системою фільтрації вмісту, навіть якщо здійснюється з-за кордону. Нещодавно дослідники виявили, що система

фільтрації розрізняє користувачів з материкового Китаю та з-за меж країни, застосовуючи різний рівень фільтрації контенту.

Додаток підпадає під китайські закони щодо цензури та перехоплення вмісту, тому дані користувача (особливо китайською) потенційно доступні державним органам КНР. У червні 2013 року індійське Бюро розвідки позначило WeChat як потенційно небезпечний додаток. В Індії обговорюється питання заборони WeChat за його можливості в зборі особистої інформації та даних від користувачів.

2.1.4 Зведена інформація про рішення

Якщо порівнювати існуючі інтерактивні системи пошуку інформації з використанням поточного місцезнаходження користувача, можна помітити що вони використовують класичні інтерфейси, мають первинною метою функції збір даних, видають упереджені дані.

Наприклад система Forsquare має метою збір інформації про об'єкти та їх деталі для подальшого продажу рекламодавцям. Відображення об'єктів та деталей про їхнє місцезнаходження винесено у окремий додаток що подає дані у ігровій формі. Неупередженість видачі пошуку об'єктів за пошукового рушію також викликає сумніви.

Система Google Maps орієнтована на відображення користувачу у першу чергу мапи. Більш того, ця система орієнтована на збір поточних місцезнаходжень користувача для подальшої їх використання у рекламних сервісах, або для продажу стороннім компаніям.

Система Facebook/Instagram шукає тільки попередньо зареєстровані та опубліковані в системі дані. Достовірність інформації покладено на осіб що публікують дані. В першу чергу видають дані що рекламуються, більш того, система потребує дозволу постійно відслідковувати місцезнаходження користувача що протиречить принципам приватності. Ця компанія та її системи також перепродають дані про місцезнаходження користувача рекламодавцям для того щоб останнім було зручніше робити так званий

таргетинг, або обмеження охоплення рекламної аудиторії у межах визначених параметрів, зокрема місцеположенням.

Всі названі компанії отримують згоду користувача на використання персональної інформації. З іншого боку без цього дозволу користувачам доступний дуже обмежений функціонал, яких, по суті, примушує погодитися надавати дані або не користуватися цими застосунками.

У той же час, компанія Alphabet, що володіє сервісами Google, має розвинуту інфраструктуру сервісів постачання даних. Один з них Google Maps Platform у складі Google Cloud.

Цей постачальник даних надає невеликий обсяг даних безкоштовно, що влаштовує вимогам академічної роботи. Але всі ці дані ніяк не відрізняються від платної версії. Також видача даних на запити не проходить ранжування, тож ці дані фактично доступні у сирому вигляді. Платформа є дуже відомою та поширеною у широкому загалі, тому існує велетенська кількість прикладів використання даних від цього провайдера.

Постачальника даних Google Maps Platform можна назвати найзручнішим у використанні на цей час.

Постачальник даних Factual використовується у пропрієтарних системах та є прикладом комерційної компанії яка купує дані у локальних постачальників даних, таких як Mapgeosystem або Visicom для України, та консолідувавши їх передає для використання стороннім компаніям. Нажаль про компанію існує мало публічних даних або порівнювань з іншими. Але сам факт того що даними цієї компанії користуються найбільші корпорації світу показує цього постачальника даних як одного з лідерів на ринку геоданих.

Доступ для академічної роботи отримати не вдалось, тож використання цього постачальника неможливо у цій роботі.

База місцеположень від компанії Here застосовуються у багатьох навігаційних системах, зокрема у автомобільній промисловості. Ця компанія також відома як покупець географічних даних у локальних провайдерів.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		22

Останні 5 років застосунки та функціональність сервісу підтримується та доповнюється спеціалістами з України, у компанії Intelias. Це дає змогу студентам, потрапивши на стажування у цю компанію, отримувати інформацію, як то кажуть, з першоджерела.

Широко розвинутий функціонал у API цього провайдера робить його серйозним конкурентом до Google Maps Platform. Водночас дані більш спрямовані на пошук та визначення шляху між точками у просторі або об'єктами. Наповнюються та оновлюються з певною затримкою, що зумовлено маленьким вкладом користувачів в розвиток цієї системи. Тож достовірність даних про об'єкти у реальному часі викликає сумніви.

Постачальних даних OpenStreetMaps – як рішення з відкритим вихідним кодом є, безперечно, найнеупередженішим постачальником даних. Розвиток цього проекту значно покладається на незалежних розробників, дані постачаються спільнотою добровільно, або державними агенціями. Це дає змогу не залежати від локальних постачальників даних та надавати вільний доступ до бази даних географічних об'єктів.

Зворотною стороною такого підходу є дуже повільне наповнювання бази даних та оновлення функцій основної гілки розробки застосунків. Так, наприклад, найсучасніше оновлення API для OpenStreetMaps датується квітнем 2009 року. Звісно, за цей час було зроблено безліч застосунків та можна знайти купу прикладів використання. Але зручність використання API, та зміст викликають питання. Швидкість відповідей не гарантована.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		23

2.1.5 Висновки до розділу

За результатами аналізу існуючих рішень з інтерактивних пошукових систем з використанням поточного місцезположення користувача можна дійти висновку що вони не відповідають вимогам безпечності та приватності, неупередженості або зручності у використанні в сучасних умовах.

Базуючись на аналізі постачальників даних можна зробити висновок що Google Maps Platform найбільш відповідає вимогам до цієї роботи.

Аналізуючи сучасні рішення що можуть бути використані як інтерактивна середа для пошукової системи я дійшов висновку що месенджери є найбільш відповідним видом середи. У сучасному світі налічується 2.71 мільярда смартфонів. Кожен з них потенційно має месенджер.

Безперечним лідером з безпечності та доступності для розробників є Telegram. Надалі в дипломному проекті буду використовуватись саме ця платформа.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		24

					ІА351.050БАК.001.ПЗ	Арк.
						25
Зм	Арк.	№ докум.	Підп.	Дата		

3. СТРУКТУРА СИСТЕМИ

Для реалізації власної інтерактивної пошукової системи з використанням поточного місцезнаходження користувача, враховуючи вимоги до системи, було обрано:

- 1) Telegram месенджер як платформу для інтерактивного спілкування з користувачем;
- 2) Telegram Bot API як спосіб комунікації інтерактивного оточення з серверною частиною;
- 3) мову програмування Python;
- 4) Google Maps Platform як постачальника даних;
- 5) операційну систему Linux для розміщення серверної частини у розподіленому оточенні сервісів DigitalOcean.

Центром системи є сервер у хмарному оточенні, з програмним забезпеченням, що розподіляє та віртуалізує ресурси між застосунками. На Діаграмі Д1 та рисунок 3.1 його позначено як Application Server.

Функціями серверу буде керувати застосунок на мові Python. Функції, які потрібно контролювати:

- 1) мережеве з'єднання з усіма учасниками процесу;
- 2) запити до постачальника даних;
- 3) запити до інтерактивного оточення;
- 4) запити до системи форматування тексту;
- 5) відповіді від вищезазначених систем;
- 6) ведення записів активності користувачів.

Функціонування системи повинно проходити наступним чином:

За допомогою запиту «/start» користувач надсилає команду до серверів TelegramAPI ініціювати або продовжити спілкування з пошуковою системою. Система TelegramAPI, в свою чергу, порівнявши всі вхідні дані від користувача, визначає до якої системи належить цей запит, та надсилає запит до серверної частини пошукової системи, що була зареєстрована раніше.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		26

Серверна частина, отримавши запит, починає свою роботу, ідентифікує користувача за допомогою даних що містяться у вхідному запиті. У відповідь формує текст, з базовою інформацією про застосування та спеціальним чином відформатовану кнопку «Надати місцеположення».

Сформований текст, та кнопку серверна частина надсилає до серверів TelegramAPI. Ця система доставляє звітку та кнопку на пристрій користувача.

Користувач відповідає на запит місцеположення. Системи визначення місцеположення на пристрої користувача отримують дані та формують відповідь, підтверджуючі попередньо сформований для нього запит. Або використовує пристрій для введення інформації для визначення свого запиту. Такими пристроями можуть бути екранна або інша клавіатуру, системи розпізнавання голосу або почерку, попередньо сформовані інтерактивні сценарії.

У цьому проекті я використовую запит «кава | coffee» у поєднанні з поточним місцеположенням користувача визначеному у термінах системи сферичних координат, а саме широти та довготи.

Далі серверна частина отримавши дані від користувача формує запит до постачальника даних. Запит являє собою спеціальним чином сформований HTTP запит який містить API Key, пошуковий запит, місцеположення, мову відповіді, та інші уточнюючі дані якщо потрібно.

Постачальник даних аналізує запит та дає відповідь серверній частині, з конкретними відомостями про об'єкти які відповідають запиту. Відповідь надається серіалізованою за допомогою JSON.

Отримавши відповідь, серверна частина десеріалізує дані та виділяє потрібне. Спеціальним чином зібрана відповідь надсилається до TelegramAPI, яка надсилає сповіщення та тіло відповіді користувачу.

На цьому цикл завершується та серверна частина починає очікувати новий запит та команду почати роботу.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		27

Запити, інформація про користувача, відповіді постачальників даних, посилання що формуються, та вся додаткова інформація для подальшого налагоджування системи може бути надана у формі виводу у консоль або записана у окремий файл, що повинен зберігатися на сервері.

Якщо приймати до уваги сучасні європейські норми та стандарти, а зокрема загальний регламент про захист даних, що діє в рамках законодавства Європейського Союзу та повинен захищати персональні дані усіх громадян ЄС, усі персональні дані повинно привести до вимог регламенту. А саме збирати дані з використанням псевдонімів чи повної анонімізації та використовувати налаштування найвищого рівня приватності за замовчуванням, так щоб дані не були доступні публічно без очевидної згоди та не могли бути використані для ідентифікації суб'єкта без додаткової інформації, що зберігається окремо. Ніякі особисті дані не можуть бути оброблені, якщо це не має під собою законних підстав, визначених регламентом, або якщо особи що мають доступ до даних не отримали явної, очевидної згоди від власника даних. Дані не повинні передаватися або оброблятися особами що не є суб'єктами ЄС.

Якщо застосунок, що підтвердив виконання регламенту про захист даних якійсь чином допустив витік, втрату, або розголошення даних, особи відповідальні за застосунок повинні упродовж сімдесяти двох годин повідомити про це наглядовому органу.

Як перший крок для відповідності вимогам регламенту про захист даних, будь яка інформація стосовно користувача повинна бути виключення із додаткової інформації для подальшого налагоджування системи одразу після введення системи у експлуатацію.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		28

4. ВИМОГИ ДО СИСТЕМИ

4.1 Налаштування Linux у розподіленому оточенні

Постачальних хмарних рішень DigitalOcean набув популярності у 2013 році, коли запропонував користувачам недорогі віртуальні машини із твердотільними накопичувачами. Завдяки цьому розгортання нового екземпляру серверної частини рішення стали дуже доступними. З доступністю почалася нова ера хмарних обчислювань. Не став винятком й цей проект.

Щоб створити нову віртуальну машину потрібно як мінімум обрати дистрибутив та версію операційної системи Linux що буде використовуватись, географічне розташування датацентру, необхідні кількість оперативної пам'яті та кількість ядер процесору. Також для створення нової машини можна використовувати інтерфейс для командного рядка. Як показано на рисунку 4.1, інтерфейс цього процесу дуже простий тож зазвичай це не приносить якихось проблем.

Таких спрощений підхід до створення нових серверів як і загальне зниження вартості процесорів, оперативної пам'яті, підвищення швидкості циклів запис/читання призвели до вибухового розвитку обчислення за допомогою самоосвітніх алгоритмів.

Тому, як і багато хто з прогресивної університетської середи, я цьому проекті використовую підхід та мови що є стандартом де факто для галузі сьогоdnішнього машинного навчання та систем штучного інтелекту.

PROJECTS

Experiments

Mozilla UA

New Project

MANAGE

Droplets

Kubernetes

Volumes

Databases

Spaces

Images

Networking

Monitoring

API

DISCOVER

Marketplace

ACCOUNT

Profile

Billing

Security

Referrals

Search by Droplet name or IP (Cmd+B)

Create

USAGE\$2.30

Create Droplets

Choose an image

Distributions

Container distributions

Marketplace

Backups

Custom images

Ubuntu18.04 x64

FreeBSD

Fedora

Debian

CentOS

Choose a plan

STARTER

PERFORMANCE

Standard

General Purpose

CPU Optimized

Standard virtual machines with a mix of memory and compute resources. Best for small projects that can handle variable levels of CPU performance, like blogs, web apps and dev/test environments.

\$5/mo\$0.007/hour

\$10/mo\$0.015/hour

\$15/mo\$0.022/hour

\$15/mo\$0.022/hour

\$15/mo\$0.022/hour

\$20/mo\$0.030/hour

1 GB / 1 CPU25 GB SSD disk1000 GB transfer

2 GB / 1 CPU50 GB SSD disk2 TB transfer

3 GB / 1 CPU60 GB SSD disk3 TB transfer

2 GB / 2 CPUs60 GB SSD disk3 TB transfer

1 GB / 3 CPUs60 GB SSD disk3 TB transfer

4 GB / 2 CPUs80 GB SSD disk4 TB transfer

Show all plans

Each Droplet adds more free data transfer to your account, starting at 1TB/month and scaling with Droplet usage and size. Additional outbound data transfer is billed at \$0.01/GB. Read more.

Add backups

Automatic system-level backups. Use the backup images to revert the server or create new Droplets. Backups cost 20% of the Droplet price.

Enable Backups

Add block storage

Currently only available in AMS3, BLR1, FRA1, LON1, NYC1, NYC3, SFO2, SGPI and TOR1.

Block storage lets you add independent storage volumes that can be accessed like local disk and moved from one Droplet to another within the same region.

Add Volume

Choose a datacenter region

New York

San Francisco

Amsterdam

Singapore

London

Frankfurt

Toronto

Bangalore

Select additional options

Private networking

IPv6

User data

Monitoring

Add your SSH keys

New SSH Key

acme

yurikhejester

Finalize and create

How many Droplets?

Deploy multiple Droplets with the same configuration.

1 Droplet

Choose a hostname

Give your Droplets an identifying name you will remember them by. Your Droplet name can only contain alphanumeric characters, dashes, and periods.

ia5tzytj.kiev.ua

Add Tags

Select project

Select an existing project for this Droplet/s.

Experiments

Create

Рисунок 4.1 – Інтерфейс створення віртуальної машини у DigitalOcean

Створення віртуальної машини на стороні постачальника займає приблизно 90 секунд. Після цього користувач отримує повністю готове оточення з обраною операційною системою. Вигляд основних відомостей про новостворену машину показана на рисунку 4.2

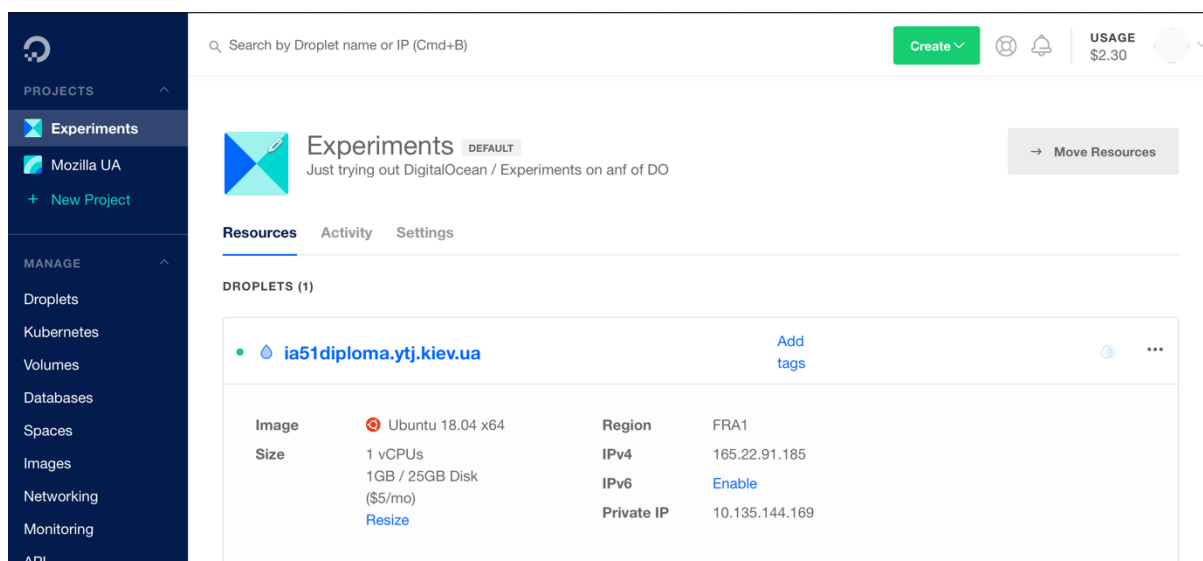


Рисунок 4.2 – Вигляд основних відомостей про систему

Якщо немає додаткових вимог щодо конфігурації мережевого або іншого обладнання, створення нових записів таких як додаткові користувачі, роботу з новоствореною операційною системою слід розпочати з оновлення всієї системи. У операційній системі Ubuntu це можна зробити з використанням команди «`apt update && apt upgrade`» від імені супер-користувача (root).

Де «`apt update`» оновить відомості про всі встановлені пакети та нові версії пакетів, а «`apt upgrade`» виконає безпосередньо оновлення всіх пакетів, встановить їх у постійну пам'ять та зробить необхідні зміни у конфігурації якщо такі потрібні.

Після оновлення системи її можна вважати готовою до роботи. Однак слід пам'ятати що із вірогідністю близькою до одиниці (The Hacker Playbook 2: Practical Guide to Penetration Testing, Volume 2) пошукові роботи що сканують інтернет вже знайшли новостворений сервер та намагаються підібрати паролі до найбільш відомих акаунтів.

Щоб уникнути такого, слід подбати про безпеку увімкнувши брандмауер, настроївши застосунки типу fail2ban, заборонити доступ користувачам за допомогою авторизації за паролем. Такі мінімальні дії суттєво знизять ризики.

4.1.1 VirtualEnv

Для того щоб зручно використовувати оточення мову Python на системах хмарних обчислень, та для того щоб застосунки можна було легко переносити із середи розробки до продуктивної середи користуються так званим «віртуальним оточенням» або «ізолюваним оточенням» для застосунків на цій мові. Таким чином всі бібліотеки та необхідні застосунку файли, в тому числі версії компіляторів, модулів, залежностей, та, частково, дозволи файлів знаходяться у одній папці, та не будуть конфліктувати з іншими проектами на сервері та інтегровані за допомогою модуля «venv».

Налаштовані таким чином оточення вже є обов'язковою умовою для використання у безсерверних хмарних оточеннях, де немає одного виділеного серверу та доступу до операційної системи, як це зроблено в межах цього проекту, а є програмне забезпечення яке приймає вже готове оточення, та сприймає кожне оточення як роздільну одиницю програмного забезпечення.

Для того щоб налаштувати ізолюване оточення потрібно встановити модуль для мови програмування Python що називається «virtualenv». Для цього рекомендовано використати вже існуючий менеджер пакетів. Це унеможливить проблеми з дозволами на використання та читання. Також можна встановити найновішу версію за допомогою наступної команди у Linux системах:

«pip install --user <https://github.com/pypa/virtualenv/tarball/master>».

Подальше налаштування потребує створення віртуального оточення за допомогою команди «virtualenv ENV» від імені користувача, що буде

використовувати застосунок, де «ENV» це ім'я директорії де повинні зберігатися файли ізолюваного оточення. Ця команда виконує наступні дії:

- 1) Створюються піддиректорії «ENV/lib» та «ENV/include» які містять файли бібліотек що підтримують нове ізолюване оточення. Також згодом сюди будуть інстальовані всі пакети для цього віртуального оточення, а конкретно до «ENV/lib/pythonX.X/site-packages/»;
- 2) створено піддиректорію «ENV/bin», тут буде розташовано ізолювану версію інтерпретатора мови Python. Це дає змогу за допомогою заголовку «#!/path/to/ENV/bin/python» запустити застосунок у ізолюваному оточенні із використанням спеціально визначеної версії «Python»;
- 3) встановлено пакети «pip» та «setuptools». Це потрібно для того щоб зручно було встановлювати бібліотеки та інші застосунки у ізолюваному оточенні.

Таким чином ми отримали нове віртуальне оточення що повністю ізолюване від версії Python яка була використана при створенні оточення.

Для введення в дію цього оточення у середовищах «bash» або «zsh» потрібно виконати команду «. /path/to/ENV/bin/activate». При цьому, буде змінено діючий параметр шляхів за замовчуванням \$PATH: до нього додається «bin/» віртуального оточення. Також буде змінено звернення командного рядка щоб однозначно вказувати яке оточення зараз використовувати.

Для того, щоб відмінити ці дії потрібно використовувати команду «deactivate».

4.1.2 Dependencies

При створенні програмного забезпечення що керує системами, дуже зручно користуватись вже готовими бібліотеками. Це зменшує час розробки застосунків та дає можливість спиратись на попередній досвід.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		33

Програмне забезпечення випускається в пакетах, таким чином, легше керувати встановленими програмами.

Менеджер пакунків являє собою набір бібліотек, які упаковані разом, що полегшує завантаження всього пакету, а не кожної бібліотеки.

Майже кожна бібліотека пакета має залежність, яку керує менеджер залежностей.

Бібліотека є набором вже написаного коду. Пакет являє собою набір бібліотек, які побудовані один на одного або використовують один одного так чи інакше.

Управління залежностями допомагає керувати всіма бібліотеками, необхідними для роботи програми. Це надзвичайно корисно, коли ви маєте справу зі складними проектами та в декількох середовищах. Управління залежностями також допомагає відслідковувати, оновлювати бібліотеки швидше і простіше, а також вирішувати проблеми, коли один пакет залежить від іншого пакета.

Кожна мова програмування має свій менеджер залежностей.

Сьогодні найбільш часто використовуваним менеджером пакунків Python є «pip», який використовується для встановлення та керування пакетами програмного забезпечення Python, що знаходяться в індексі пакета Python. Pip допомагає без особливих зусиль «вручну» контролювати встановлення та життєвий цикл публічно доступних пакунків Python з їхніх онлайн-сховищ. Pip також може оновлювати, показувати, видаляти залежності від проекту тощо.

Щоб встановити пакунок, потрібно запустити програму `pip install <somepackage>`, яка створить додаткову бібліотеку Python у віртуальному середовищі, або домашньому каталозі.

Запуск `pip freeze`, може допомогти перевірити встановлені пакунки та версії пакетів, перелічені у відсортованому за регістром порядку.

Використані бібліотеки, та опис призначення наведено у таблиці 4.1.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		34

Таблиця 4.1 – опис використаних бібліотек

Назва бібліотеки	Призначення
time	бібліотека доступна у мові Python за замовчуванням. Цей модуль забезпечує різні функції, пов'язані з часом.
logging	бібліотека доступна у мові Python за замовчуванням. Готовий до використання і потужний модуль для того щоб створити однорідний журнал дій для застосунка.

Продовження таблиці 4.1

Назва бібліотеки	Призначення
python-telegram-bot	ця бібліотека забезпечує чистий інтерфейс Python для програмного забезпечення Telegram Bot. Він сумісний з версіями Python 2.7, 3.3+ і PyPy. На додаток до чистої реалізації API, ця бібліотека має ряд класів високого рівня, які роблять розробку ботів простим і простим. Ці класи містяться в підмодулі телеграми.ext.
googlemaps	Клієнт Python для служб "Карти Google" - це бібліотека клієнтів Python для таких API Карт Google: Directions API, Distance Matrix API, Elevation API, Geocoding API, Geolocation API, Time Zone API, Roads API, Places API
urllib3	urllib3 приносить багато критичних функцій що відсутні за замовчанням, таких як: <ol style="list-style-type: none"> 1) Безпека ниток застосунку 1) Пул з'єднань. 2) Перевірка SSL / TLS на стороні клієнта. 3) Завантаження файлу з багатошаровим кодуванням. 4) Помічники для повторної спроби та переадресація HTTP. 5) Підтримка кодування gzip та deflate. 6) Підтримка проксі для HTTP і SOCKS.

Продовження таблиці 4.1

Назва бібліотеки	Призначення
emoji	бібліотека що дозволяє перетворювати емодзі коди на символи юнікоду для безпечного форматування тексту на всіх системах.
georu	геору дозволяє розробникам Python легко знаходити координати адрес, міст, країн і орієнтирів по всьому світу за допомогою геокодерів третіх сторін та інших джерел даних. Також використовується для маніпуляцій з координатами. Дозволяє обчислювати геодезичну відстань між двома точками, використовуючи геодезичну відстань або відстань великого кола. Відстань великого кола (great_circle) використовує сферичну модель Землі, використовуючи середній радіус Землі, як визначено Міжнародним союзом геодезії та геофізики, $(2a + b) / 3 = 6371.0087714150598$ кілометрів приблизно 6371.009 км (для WGS-84) , що призводить до похибки до приблизно 0,5%.

4.2 Google Maps Platform як постачальних даних

За даними компанії у Google Maps Platform зібрали дані про 99% куточків світу. Вони зазначають що обробляють 25 мільйонів оновлень існуючої інформації від 1000 різноманітних джерел щодня.

Взагалі на платформі існує три продукти:

- 1) Maps;
- 2) Routes;
- 3) Places.

Продукт під назвою «Mars» - це основний продукт на платформі. Він дозволяє отримувати візуальну картографічну інформацію по запиту. Також можна написати свої карти за допомогою спеціальних маркерів, ліній, кольорів, багатокутників і зображень. Надати користувачам можливість створювати та спільно використовувати власні карти та використовувати масштабування, стискання, обертання та нахил для більш детального вивчення карт. Ви можете виділити розташування магазинів за допомогою спеціальних кольорів та елементів. Або налаштувати віртуальний велосипедний шлях за допомогою функції перегляду вулиць за допомогою унікальних маркерів, накладок і фотографій.

Коли ви будете на тій же інфраструктурі, що обслуговує мільярди користувачів Карт Google, ви можете розраховувати на платформу, яка масштабуватиметься та змінюється, на вашу вимогу. На цій безпечній інфраструктурі, що підтримує майбутнє, можна зростати від прототипу до масштабу планети без необхідності думати про потужність, надійність або продуктивність.

Складається з двох компонентів: «Mars» - безпосередньо мапи з маркерами, лініями, полігонами, прямими, тощо; та «Street View» - панорамні фото з вулиць або приміщень що збираються компанією.

Продукт під назвою «Routes» - це додатковий продукт який дозволяє обчислити шлях до місця призначення, враховуючі ваші можливості, транспорт, поточні умови, дані трафіку у реальному часі. Підтримує 25 точок на маршруті. Використовує штучний інтелект на системи самонавчання для надання прогнозів щодо майбутнього стану трафіку на дорогах для легкого планування маршрутів.

Продукт під назвою «Places» - це додатковий продукт який дозволяє надавати користувачам дані про назви місцеположень, адреси, рейтинги, відгуки, контактну інформацію та атмосферу. Місцеві гідни та користувачі подають десятки мільйонів оновлень щодня. Містить інформацію про більш ніж 150 мільйонів місць на земній кулі.

					IA351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		38

У цьому проекті використані продукти Places та Routes. За допомогою Google Places API серверна сторона отримує дані про об'єкти що підпадають під запит користувача. Через Google Routes API – серверна сторона формує спеціальним чином підготовлене посилання, відкривши яке користувач зможе скористатись маршрутом у додатку Google Maps

4.2.1 Google Places API Request

Функції в бібліотеці служб Places API дозволяють програмі шукати місця (визначені в цьому API як установи, географічні місця або визначні місця), що містяться в певній області, наприклад, в межах карти або навколо фіксованою точкою.

API служби Places на карті надає функцію автозаповнення, яку можна використовувати, щоб надати своїм програмам поведінку пошуку наперед у полі пошуку Карт Google. Коли користувач починає вводити адресу, автозаповнення залишить решту.

Перш ніж використовувати бібліотеку API служби Places, спочатку треба ввімкнути її в консолі Google Cloud Platform.

Як виглядає сторінка керування API у консолі Google Cloud Platform зафіксовано на рисунку 4.4

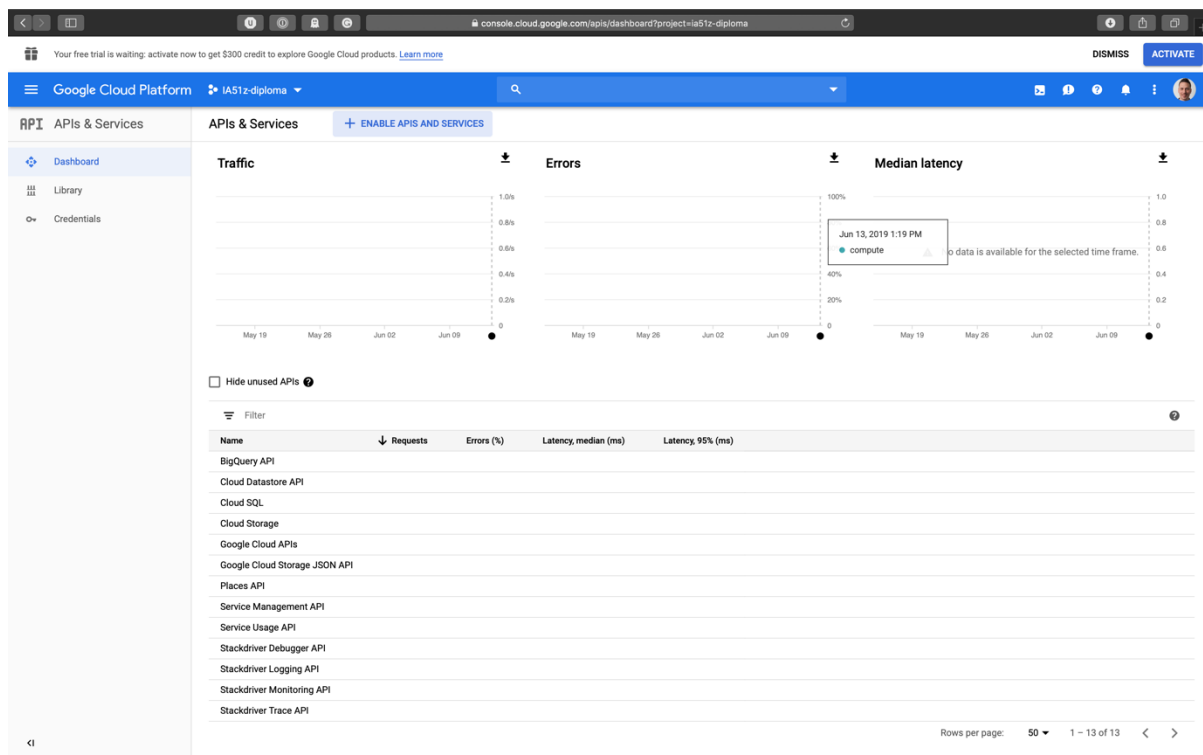


Рисунок 4.4 – Консоль Google Cloud Platform

У службі підтримуються різні методи доступу до даних. Одним з них є додаткова бібліотека що встановлюється з серверної сторони. Існують реалізації для різних мов програмування таких як JavaScript, Python, GoLang, Node.js, Objective-C, Java. У моєму випадку використовується модуль для мови Python що розробляється та підтримується силами спільноти: <https://github.com/googlemaps/google-maps-services-python>

За використанням цього модуля слід використовувати наступні інструкції для пошуку місць за допомогою API служби Places.

Запит пошуку місця (Find Place request).

Запит пошуку місця приймає на вхід текстове введення та повертає місце. Текс може бути будь-якою частиною з деталей об'єкту, наприклад ім'я, адреса, частина адреси, або номер телефону

Запит пошуку місця це HTTP URL строгої форми, де HTTP – це протокол сьомого, або прикладного рівня моделі OSI [3], а URL – це

уніфікований локатор ресурсів, або адреса ресурсу у мережі інтернет у вигляді тексту, та виглядає наступним чином:

<https://maps.googleapis.com/maps/api/place/findplacefromtext/output?parameters>

У відповідності до стандарту URL усі параметри повинні бути розділені знаком амперсанд, що відображається символом «&».

Відповідь на такий запит може бути серіалізований у форматі JSON, що рекомендовано, або XML.

Деякі параметри обов'язково повинні буди у запиті щоб ініціювати запит пошуку місця, а саме:

- 1) key - ключ для використання Google Cloud Platform API . Цей ключ ідентифікує застосунок;
- 2) input -Текстове введення що визначає яке місце шукати, наприклад ім'я, адреса, або номер телефону;
- 3) inputtype – тип текстового введення Може приймати значення textquery або phonenumber. Якщо введення використовує номер телефону, то номер телефону повинен бути вказаний у міжнародному форматі, у відповідності до стандарту E.164 ITU.

Додаткові параметри запиту місця:

- 1) language – код мови, що вказує якою мовою пошукові результати повинні бути повернуті, якщо це можливо. Також пошуковий рушій приймає цей параметр як важливий та результати які містять вказану мову можуть мати вищий рейтинг ніж результати іншою мовою. Для застосування української мови слід використовувати значення «uk»;
- 2) fields – які конкретно поля з усіх даних про об'єкт повинні буди повернуті. Значення повинно бути списком розділеним комами;
- 3) locationbias – визначає місце та пріоритет пошуку, можна вказувати географічні координати широти та довготи та радіус пошуку, або можна вказати дві пари широти та довготи вказуючи прямокутник де слід шукати. Якщо цей параметр не вказано, API буде використовувати IP адресу емітента запиту як параметр за замовчуванням

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		41

Визначення locationbias:

- IP bias: надає інструкцію API використовувати IP адресу для визначення місця пошуку. Передає строку ipbias, не містить додаткових параметрів.
- Point: одиночне місцеположення, складається з пари шіроты, довгота. Повинно використовуватись у форматі «point:lat,lng».
- Circular: строковий параметр що вказує на радіус пошуку у метрах додатково до географічних точок. Повинно використовуватись у форматі: «circle:radius@lat,lng».
- Rectangular: строковий параметр що вказує на прямокутник місця пошуку. Повинен складатись з пари географічних координат з точністю до десятої градуса, представляючі південь/захід та північ/схід прямокутника. Повинно використовуватись у форматі: «rectangle:south,west|north,east». Потрібно зауважити що значення східного/західного рівня обертаються до діапазону -180, 180, а значення північ/південь притискаються до діапазону -90, 90.
- Якщо пропустити цей параметр у запиті пошуку місця, тільки параметр place_id може бути повернений у відповіді.
- Fields: повинно бути використано для повернення тільки потрібних полів у відповіді на запит. Повинно використовуватись у форматі: fields=opening_hours,icon,geometry. Потрібно використовувати косу риску при визначенні складних значень. Наприклад: geometry/location.

Поля відповідають результатам пошуку місця і поділяються на три категорії виставлення рахунків Basic, Contact, та Atmosphere. Основні поля (Basic) тарифікуються за базовою ставкою і не несуть додаткових витрат. Поля для контактів і атмосфери тарифікуються за більш високою ставкою. Для отримання додаткової інформації див. Атрибути (html_attributions) завжди повертаються при кожному виклику, незалежно від того, чи

					IA351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		42

запитувалося поле. Застереження: запити пошуку місця та запити деталей місця не повертають ті ж поля. запити пошуку місця повертають підмножину полів, які повертаються за запитами деталі місця. Якщо потрібне поле не повертається пошуком місця, можна скористатися функцією пошуку місця, щоб отримати place_id, а потім використати цей ідентифікатор місця, щоб зробити запит на деталі місця.

Категорія основні поля (Basic) містить у собі наступне:

formatted_address, geometry, icon,name, permanently_closed, photos, place_id,plus_code, types.

Категорія контакти (Contacts) включає у себе наступні поля: opening_hours.

Запит пошуку місця повертає тільки чи відкрито місце у момент запиту. Для того щоб отримувати повну інформацію слід Запити деталей місця.

Категорія атмосфера (Atmosphere) включає в себе: price_level, rating, user_ratings_total. Іншими словами показує рейтинг місця.

Наступний приклад показує Запит пошуку місця "Музей сучасного мистецтва Австралії", включно з фото, адресом, рейтингом, ім'ям, годинами роботи та геометрією:

«https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=Museum%20of%20Contemporary%20Art%20Australia&inputtype=textquery&fields=photos,formatted_address,name,rating,opening_hours,geometry&key=YOUR_API_KEY»

Наступний приклад показує Запит пошуку місця для "Монгольського гриля", використовуючи параметр locationbias, щоб віддати перевагу результатам у межах 2000 метрів від вказаних координат:

«https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=mongolian%20grill&inputtype=textquery&fields=photos,formatted_address,name,opening_hours,rating&locationbias=circle:2000@47.6918452,-122.2226413&key=YOUR_API_KEY»

У наступному прикладі показано Запит пошуку місця з використанням номер телефону. Варто звернути увагу, що префікс міжнародного виклику

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		43

"+" був закодований у %2B, тому цей запит є сумісною URL-адресою.

Залишившись без кодування, префікс + буде декодований на простір на сервері, що призведе до неправильного пошуку номера телефону:

«<https://maps.googleapis.com/maps/api/place/>

[findplacefromtext/json?input=%2B63804566700&inputtype=phonenumbers&fields=place_id&key=YOUR_API_KEY](https://maps.googleapis.com/maps/api/place/findplacefromtext/json?input=%2B63804566700&inputtype=phonenumbers&fields=place_id&key=YOUR_API_KEY)»

Запиту пошуку місця неподалік (Nearby Search Request)

Запит пошуку місця неподалік та Текстовий запит повертають всі доступні відомості про вибрану місцевість (підмножину даних полив що підтримуються). Обидва ці запити не можуть бути обмежені по переліком полів що повертаються. Для того щоб отримувати тільки поля що потрібні користувачам, треба використовувати звичайний Запит пошуку місця.

Запит пошуку місця неподалік вказує пошуковому рушію шукати місця у визначеній області. Для того щоб уточнити пошуковий запит можливо використовувати ключові слова або тип місця що потрібно знайти.

Запит пошуку місця неподалік це HTTP URL наступної форми:
«<https://maps.googleapis.com/maps/api/place/nearbysearch/output?parameters>»

У відповідності до стандарту URL усі параметри повинні бути розділені знаком амперсанд, що відображається символом «&».

Деякі параметри обов'язково повинні буди у запиті щоб ініціювати запит пошуку місця, а саме:

- 1) key - ключ для використання Google Cloud Platform API . Цей ключ ідентифікує застосунок;
- 2) location - географічні координати місцеположення об'єкту. Обов'язково повинно містити широту та довготу точки.
- 3) radius – визначає дистанцію пошуку від точки у метрах. Максимально дозволений радіус 50 000 метрів.

Додаткові параметри Запиту пошуку місця неподалік

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		44

- 1) keyword – ключове слово, яке співпадає з вмістом, індексованим Google для цього місця, включаючи, але не обмежуючись назвою, типом та адресою, а також відгуками клієнтів та іншим вмістом третіх сторін.
- 2) language - код мови, що вказує якою мовою пошукові результати повинні бути повернуті, якщо це можливо. Також пошуковий рушій приймає цей параметр як важливий та результати які містять вказану мову можуть мати вищий рейтинг ніж результати іншою мовою. Для застосування української мови слід використовувати значення «uk».
- 3) name – термін що співпадає з вмістом індексованим Google для цього місця. Еквівалент «keyword». Не повинно бути тільки назвою місця. Значення в цьому полі поєднуються зі значеннями в полі ключового слова і передаються як частина одного і того ж рядка пошуку. Рекомендовано використовувати тільки «keyword» параметр для пошуку.
- 4) opennow – вказує пошуковому рушію повертати тільки ті місця що відкриті у час коли було прийнято запит. Місця що не вказують години роботи в базі даних Google Places не будуть додані у відповіді на запит у разі використання цього параметру.
- 5) rankby – визначає порядок в якому результати будуть повернуті.
Можливі значення:
 - prominence (за замочанням) - цей варіант сортує результати на основі їх важливості. Рейтинг надаватиме перевагу визначним місцям у зазначеній області. На характерне місце може вплинути рейтинг місця в індексі Google, глобальна популярність та інші фактори;
 - distance - ця опція змінює результати пошуку у порядку зростання по відстані від вказаного місця. Якщо вказано відстань, потрібно вказати одне або декілька ключових слів, назви чи типу;
- 6) type — Обмежує результати до місць, що відповідають вказаному типу.
Можна вказати лише один тип.

Наступний приклад - це запит на пошук місць типу "ресторан" у радіусі 1500 метрів точки в Сіднеї, Австралія, що містить слово "круїз":
https://maps.googleapis.com/maps/api/place/nearbysearch/json?location=-33.8670522,151.1957362&radius=1500&type=restaurant&keyword=cruise&key=YOUR_API_KEY

4.2.2 Google Places API Reply

Відповідь на Запит пошуку місця містить елементи «status», «results», «html_attributions», та «next_page_token»

Швидко визначити статус запиту дозволяють коди статусу з елементу «status». Також відповідь може містити додаткову інформацію для наступного налагоджування. Google Places API може повернути наступні статуси кодів у своїй відповіді:

- ZERO_RESULTS - вказує на те, що пошук був успішним, але не отримав результату. Це може статися, якщо до пошуку передано координати у віддаленому місці де немає об'єктів.
- OVER_QUERY_LIMIT- Вказує на те, що квота використання Google API використана.
- REQUEST_DENIED - вказує, що ваш запит відхилено, як правило, через відсутність недійсного параметра ключа.
- INVALID_REQUEST - зазвичай вказує, що необхідний параметр запиту (розташування або радіус) відсутній.
- UNKNOWN_ERROR- вказує на помилку на стороні сервера; якщо повторити спробу, результат може бути успішним.

Коли Google Place API повертає результати Запиту пошуку місця, їх розміщено у масиві даних «results». Навіть якщо результат пошуку не буде містити об'єктів (наприклад коли вказане місцезнаходження дуже віддалене від будь яких об'єктів) все одно буде повернутий пустий масив даних.

Кожен елемент масиву даних «results» містить один об'єкт з вказаної області пошуку (координати та радіус) from the specified area (location and radius), впорядковані за популярністю.

Додатково результат може містити додаткові відомості про об'єкт. Наприклад так виглядають додаткові відомості про об'єкт у форматі JSON:

```
"html_attributions" : [
```

```
"Listings by \u003ca href=\"http://www.example.com/\" \u003eExample  
Company\u003c/a\u003e"
```

```
],
```

Кожен об'єкт з масиву результатів може містити наступні поля:

- 1) icon – адреса (URL) рекомендованої іконки для відображення користувачу;
 - 2) geometry – містить геометричні параметри об'єкту, звичайно це включаючи;
 - 3) global_code – чотири символи з коду місцевості та шість символів або більше локального коду ('9G2GCFQH+H8');
 - 4) compound_code – шість символів або більше з локальним кодом місцевості та is a 6 character or longer local code with an явними даними про місцевість ('CFQH+H8 Київ, місто Київ, Україна');
- Як правило повертається «global_code» та «compound_code»; однак, якщо об'єкт знаходиться у віддаленій місцевості (наприклад океан, або пустеля) тільки «global_code» буде повернуто;
- 5) name – містить ім'я що читається людиною для повернутого об'єкту; для об'єктів господарювання зазвичай це назва компанії.
 - 6) opening_hours, містить значення open_now - що вказує, чи місце відкрите в поточний час.
 - 7) photos[] — масив даних фотографічних відомостей про об'єкт, кожен запис містить посилання на. Зазвичай повертається тільки одне посилання; для того щоб отримати додаткові посилання на світлини потрібно робити запит додаткової інформації про об'єкт.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		47

- 8) `place_id` — текстовий унікальний ідентифікатор об'єкту що вказує на місце; щоб отримати додаткові відомості про об'єкт, треба передати цей ідентифікатор у параметрі «`placeId`» розширеного Запиту відомостей про об'єкт.
- 9) `price_level` — вказує ціновий рівень для знайденого об'єкту від 0 до 4. Точні відомості можуть відрізняти для різних місцевостей. Ціновий рівень може приймати наступні значення:
- 0 — Безкоштовно
 - 1 — Недорого
 - 2 — Середньо
 - 3 — Дорого
 - 4 — Дуже дорого
- 10) `rating` – містить рейтинг знайденого об'єкту від 1.0 до 5.0, базуючись на оцінках користувачів Google Maps.
- 11) `vicinity` - містить назву об'єкта поблизу. Часто ця функція відноситься до вулиці або околиці в межах даних результатів. Цей параметр повертається лише для Запиту пошуку місця неподалік.
- 12) `formatted_address` – це строка що містить адресу об'єкту у форматі що може бути прочитана. Зазвичай це поштова адреса об'єкту.
- 13) `permanently_closed` – це флаг що вказує чи цей об'єкт закрито назавжди. Якщо місце відкрите – поле не попадає у відповідь.

4.2.3 Google Maps URLs

Важливою частиною запропонованого рішення є подальша взаємодія з оточенням у пристрої користувача. Такою взаємодією може бути відкриття мапи та формування маршруту до об'єкту. Для цього можна спеціальним чином сформувати посилання на мапу Google і таким чином відкрити застосунок Google Maps, або відповідну веб-сторінку.

Для того щоб сформувати посилання яке запустить пошук та формування маршруту у застосунку, потрібно виконати наступні правила.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		48

Посилання повинне буди наступного формату:

<https://www.google.com/maps/dir/?api=1¶meters>

Де «parameters» можуть містити:

- 1) origin: визначає початкову точку, з якої слід відображати напрямки.
Налаштування за умовчанням для найбільш релевантного початкового місцезнаходження, наприклад, розташування користувача, якщо це можливо. Якщо ні, отримана карта може надавати пусту форму, щоб дозволити користувачеві ввести початок. Значенням може бути ім'я місця, адреса або координати широти / довготи, розділені комами. Рядок має бути вилучено URL-адресою, тому адресу, наприклад, "City Hall, New York, NY", слід перетворити на City + Hall% 2C + New + York% 2C + NY;
- 2) destination: визначає кінцеву точку напрямків. Якщо ні, отримана карта може надавати порожню форму, щоб дозволити користувачеві ввести призначення. Значенням може бути ім'я місця, адреса або координати широти / довготи, розділені комами. Рядок має бути вилучено URL-адресою, тому адресу, наприклад, "City Hall, New York, NY", слід перетворити на City + Hall% 2C + New + York% 2C + NY;
- 3) destination_place_id (необов'язково): ідентифікатор місця - це текстовий ідентифікатор, який унікально ідентифікує місце. Якщо ви намагаєтеся остаточно вказати заклад, використання ідентифікатора місця є найкращою гарантією того, що ви посилаетесь на потрібне місце. URL-адреси, які використовують цей параметр, також повинні включати призначення;
- 4) travelmode (необов'язково): Визначає спосіб подорожі. Варіанти їзди, ходьби (що надає перевагу пішохідним доріжкам і тротуарам, де є), їзда на велосипеді (які маршрути по велосипедних доріжках і вибрані вулиці, де є) або транзит. Якщо не вказано жодного режиму подорожі, карта Google показує один або декілька найбільш релевантних режимів для зазначеного маршруту та / або налаштувань користувача.

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		49

Таким чином вже сформоване посилання буде мати наступний вигляд:
https://www.google.com/maps/dir/?api=1&destination=%D0%9F%D0%BE%D0%B2%D1%96%D1%82%D1%80%D0%BE%D1%84%D0%BB%D0%BE%D1%82%D1%81%D1%8C%D0%BA%D0%B8%D0%B9+%D0%BF%D1%80%D0%BE%D1%81%D0%BF%D0%B5%D0%BA%D1%82%2C+11%2C+%D0%9A%D0%B8%D1%97%D0%B2&destination_place_id=ChIJwzEIk5PO1EARSrMHEQ3z2fw&travelmode=walking

4.3 Bot API та Telegram Messenger як інтерактивне оточення

Боти - це програми різних напрямленостей використання, які виконуються всередині месенджеру. В даному контексті, всередині месенджеру Telegram. Користувачі можуть взаємодіяти з ботами, надсилаючи їм повідомлення, команди та вбудовані запити. Керування ботами виконуються за допомогою допомогою HTTPS-запитів до Telegram Bot API.

В основі програми Telegram Bots лежать спеціальні облікові записи, які не вимагають додаткового номера телефону. Користувачі можуть взаємодіяти з роботами двома способами:

Надсилати повідомлення та команди ботам, відкриваючи чат з ними або додаючи їх до груп. Це корисно для роботів чату або новинних ботів, наприклад як офіційний бот TechCrunch.

Надсилати запити безпосередньо з поля введення, ввівши @username і запит. Це дозволяє надсилати вміст із вбудованих ботів безпосередньо в будь-який чат, групу або канал.

Повідомлення, команди та запити, надіслані користувачами, передаються до програмного забезпечення, запущеного на серверах власників ботів. Таким чином, посередницький сервер, як показано на Діаграмі Д1, обробляє все шифрування і зв'язок з API Telegram. Вони спілкуються з цим сервером через простий інтерфейс HTTPS, який пропонує спрощену версію API Telegram. Цей інтерфейс називають.

					ІА351.050БАК.001.ПЗ	Арк.
						50
Зм	Арк.	№ докум.	Підп.	Дата		

Різниця між акаунтами живих людей та акаунтами ботів полягає у наступному:

- 1) боти не мають он-лайн-статусу і останніх побачених тимчасових міток, інтерфейс показує мітку "бот";
- 2) боти мають обмежене зберігання в хмарі - старі повідомлення можуть бути видалені сервером незабаром після їх обробки;
- 3) боти не можуть ініціювати розмови з користувачами. Користувач повинен або додати їх до групи, або спочатку надіслати їм повідомлення. Люди можуть використовувати посилання telegram.me/<bot_username> або ім'я користувача, щоб знайти свого бота;
- 4) Імена користувачів Bot завжди закінчуються символом "бот" (наприклад, @TriviaBot, @GitHub_bot, @ia35105_bot);
- 5) коли додано до групи, боти не отримують всі повідомлення за замовчуванням;
- 6) боти ніколи не їдять, не сплять або скаржаться (якщо явно не запрограмоване інакше).

Боти являють собою сучасну та зручну форму інтерактивної комунікації між людиною та інформаційними системами. Такими системами зараз можуть виступати пошукові рушії, системи продажу товарів або даних, персональні помічники, асистенти з комунікаційних питань на кшталт отримання та уточнення простих запитів від користувача.

4.3.1 Створення Telegram Bot

Для того щоб створити нового бота в месенджері Telegram потрібно надіслати такий запит до спеціального боту у середині служби, який зветься Telegram Bot Father (Батько ботів Телеграм). Приклад такого запиту вказано на рисунку 4.5

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		51

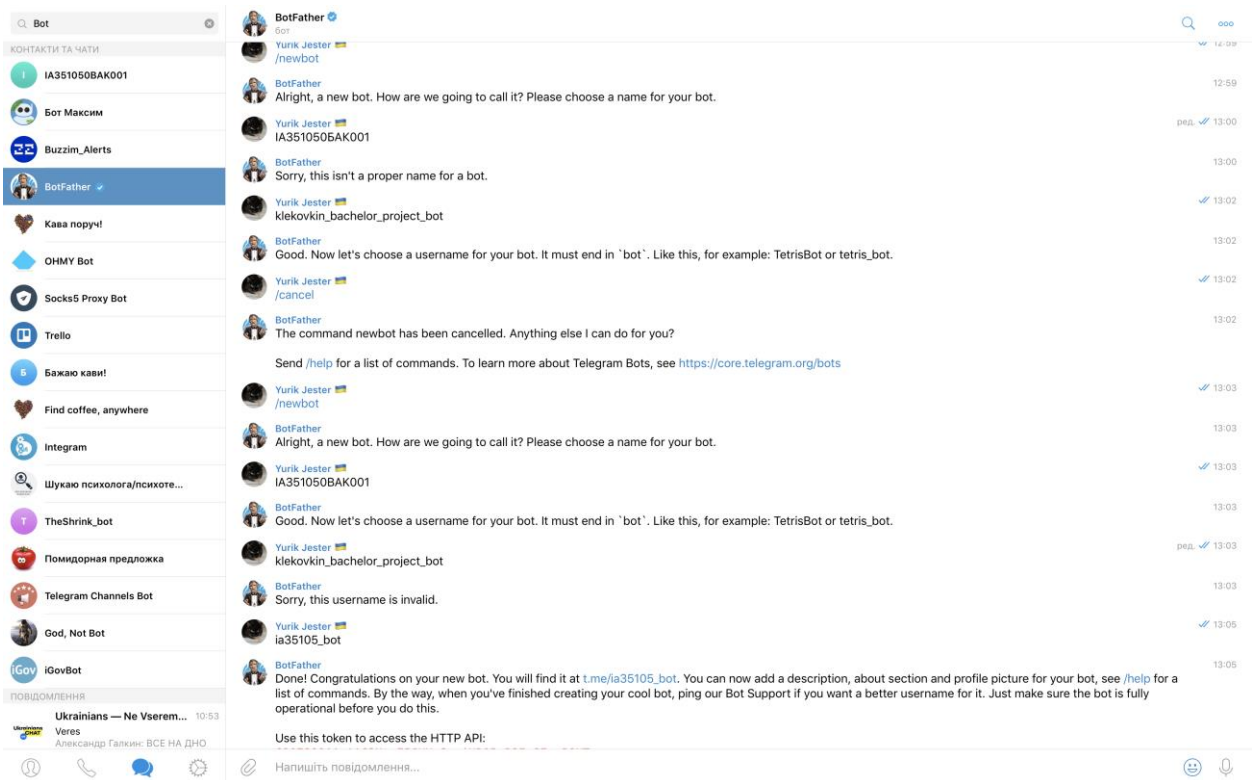


Рисунок 4.5 – Створення Telegram Bot

4.3.2 Запит координат

Для запиту координат використовується функція з Bot API та спеціальним чином форматована кнопка.

Для того щоб відобразити кнопку використовується об'єкт Bot API з назвою `telegram.KeyboardButton`. Цей об'єкт репрезентує спеціального виду клавіатуру що містить у собі одну клавішу. Об'єкт має наступні властивості:

`text` – текст кнопки;

`request_contact` – опціонально, якщо потрібно взяти номер телефона користувача;

`request_location` – опціонально, якщо потрібно надіслати місцезнаходження користувача.

Таким чином ми отримуємо наступний код для генерації кнопки:

```
share_location_keyboard = telegram.KeyboardButton(text='ПОДІЛИТИСЯ РОЗТАШУВАННЯМ', request_location=True)
```

Сервери Telegram Bot API отримавши через бібліотеку `python-telegram-bot` запит про надання розташування надсилають такий запит на пристрій

					ІА351.050БАК.001.ПЗ	Арк.
						52
Зм	Арк.	№ докум.	Підп.	Дата		

користувача. Цей запит повинен бути опрацьований багатьма системами пристрою користувача. На прикладі телефону на базі iOS (iPhone X) можна позначити що використовуються системи:

- 1) інфрачервоне випромінювання для того щоб підсвітити об'єкт розпізнавання;
- 2) лазерне випромінювання для розмітки об'єкту розпізнавання;
- 3) розпізнавання образів, для розблокування телефону;
- 4) відображення відео сигналів на дисплеї;
- 5) збір вводу від користувача з використанням сенсорів;
- 6) отримавши згоду користувача знайти його місцеположення за допомогою модулів GPS, залучивши додаткову інформацію про базові станції;
- 7) надіслати інформацію використовуючи мережеві модулі та технології.

Тож не дивлячись на поверхневу простоту використання, маленький запит потребує злагодженої роботи багатьох систем та модулів у мобільних або інших пристроях.

4.3.3 Надсилання повідомлення

Для того щоб надіслати користувачу повідомлення, потрібно певним чином сформулювати повідомлення використовуючи методи з Telegram Bots API, з додаванням форматування у форматі Markdown або HTML, з додаванням сучасних елементів спілкування – емодзі.

Щоб відправити текстове повідомлення використовується метод API з назвою Telegram.Message. Цей об'єкт має велику низку різноманітних об'єктів. В контексті цього проекту використовується об'єкт «reply_text».

Цей об'єкт повинен містити тіло повідомлення та може містити опції такі як стан екранної клавіатури, та інше.

Форматування Markdown це полегшена мова розмітки даних, створена щоб спростити та зробити форматування текстів зручнішим. Багато ідей для цього стилю форматування позичено з існуючих домовленостей у розмітці

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		53

тексту в електронних листах. Наприклад для того щоб зробити жирний текст потрібно використати оточення зірочками з обох боків виділеної частини тексту. Першу реалізацію Markdown написано Грубером на Perl, але з часом з'явилися багато реалізацій від сторонніх розробників (див. нижче). Реалізація на Perl розповсюджується за ліцензією BSD. Сайти типу GitHub, Reddit та Stack Overflow використовують Markdown для полегшення форматування користувацьких коментаріїв.

Форматування HTML це стандарт розмітки веб-сторінок та застосунків, текстів. Цей спосіб форматування описує структури веб-сторінок семантично, де включені об'єкти-сигнали для відображення браузером сторінок. Частиною стандарту – є форматування текстів. Наприклад щоб зробити текст жирним, потрібно використати оточення виділеної частини тексту об'єктами ` text `.

Емодзі являють собою стандартизовані за назвою та відображенням символи. Стандарт встановлює консорціум Unicode та дозволяє людям у всьому світі використовувати комп'ютери будь-якою мовою. Вільно доступні специфікації та дані є основою для інтернаціоналізації програмного забезпечення у всіх основних операційних системах, пошукових системах, додатках та застосунках. Важливою частиною місії консорціуму Unicode є виховання та залучення наукових та наукових кіл, громадськості.

Якщо об'єднати усі ці відомості, то для відправки повідомлення ми маємо наступну конструкцію:

```
# create a text message with emoji
```

```
message = emoji.emojize(
```

```
    f'Привіт, {user.first_name} :wave::grey_exclamation:\nДопоможи із  
захистом студенту Клековкіну\nнадішли своє *розташування*\nпта дізнайся  
де *випити кави* :coffee: поруч :blush:\n:point_down:', use_aliases=True)
```

```
# send message including 'SHARE LOCATION' keyboard
```

```
update.message.reply_text(message, reply_markup=markup, parse_mode=  
telegram.Parse Mode.MARKDOWN)
```

					ІА351.050БАК.001.ПЗ	Арк.
Зм	Арк.	№ докум.	Підп.	Дата		54

Додатково можна додати у повідомлення кнопку, що буде містити у собі посилання та текстову інформацію про місце з видачі пошукового рушія постачальника даних. Для цього слід використати об'єкт `InlineKeyboardButton` з Telegram Bot API. Цей об'єкт повинен містити у собі текст, та не більш ніж один опціональний параметр який може приймати наступні значення:

- 1) `url` - HTTP URL який буде відкрито при натисканні на кнопку;
- 2) `callback_data` – при натисканні дані будуть відправлені у запиті «callback» до серверної частини може містити від 1 до 64 байтів;
- 3) `switch_inline_query` – надішле запит до користувача відкрити список з його чатів, вибрати один чат, та додати ім'я бота та додатковий запит до поля вводу;
- 4) `switch_inline_query_current_chat`- додасть ім'я бота та додатковий запит до поля вообду у тому ж чаті що зараз використовується.
- 5) `pay` – якщо вказано, по натисканню надішле запит до серверів телеграм виконати функцію «Сплатити» та показати відповідний діалог користувачу.

Таким чином для отримання кнопки використовується наступна конструкція:

```
inline_button_text = emoji.emojize(
    f'{distance} метрів :runner: йти по Гугломапі', use_aliases=True)
update.message.reply_text(message,
reply_markup=InlineKeyboardMarkup(
    [[InlineKeyboardButton((inline_button_text), url=url)]]),
parse_mode=telegram.ParseMode.MARKDOWN)
```

Де параметр «distance» це обчислена за допомогою бібліотеки геору відстань до об'єкту.

5.ПОСІБНИК КОРИСТУВАЧА

Для того щоб почати роботу інтерактивної пошукової системи потрібно написати на надіслати повідомлення до системи із вмістом «/start», або натиснути відповіде посилання у вікні месенджера. Стартову сторінку представлено на рисунку 5.1 у лівій стороні. Після отримання такого повідомлення система відповідає з короткою інструкцією та кнопкою «поділитися розташуванням». Після натискання на цю кнопку, як показано на рисунку 5.1 у правій стороні, система мобільного пристрою отримавши запит поточного розташування користувача вимагає додатково підтвердити надсилання даних.

Після отримання місцеположення, як показано на рисунку 5.2 ліва сторона, система відповість із списком знайдених об'єктів, їх рейтинг, відстань до об'єкту, посилання щоб відкрити посилання на маршрут до об'єкту, та посилання розпочати знов.

Якщо система не знайде ніяких об'єктів – користувач отримає повідомлення про це та посилання розпочати знов, як показано на рисунку 5.2 права сторона.

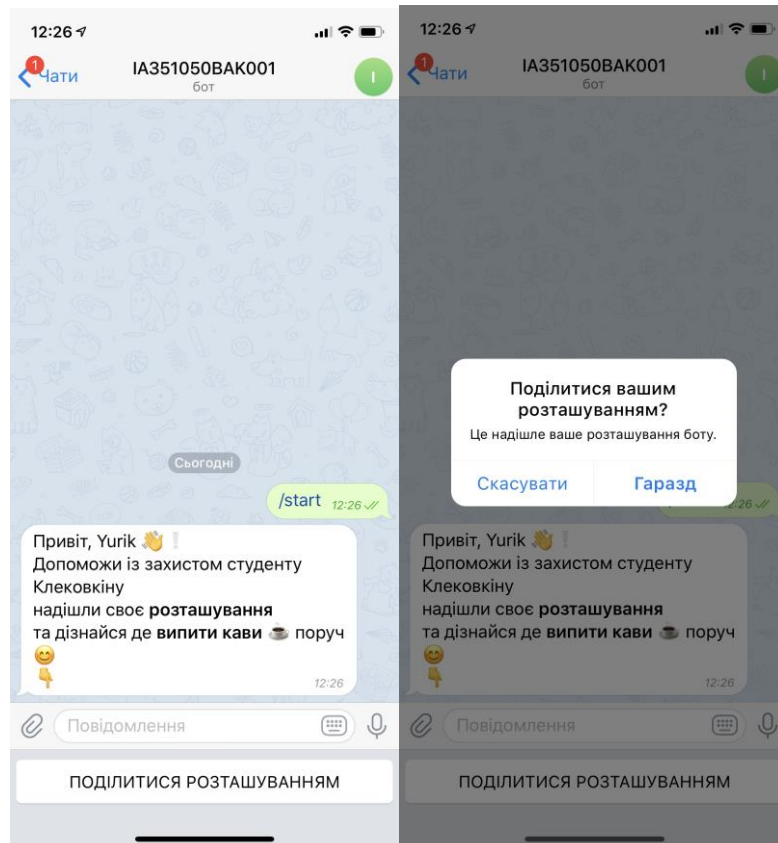


Рисунок 5.1 – початок роботи

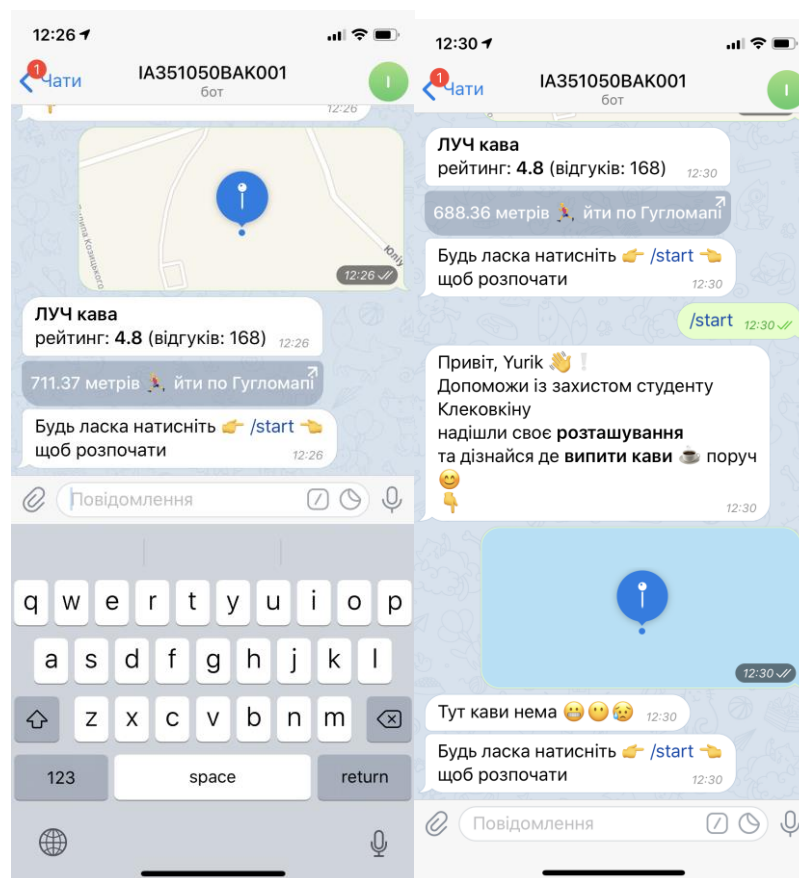


Рисунок 5.2 – отримання результатів

6. ВИСНОВКИ

У дипломному проекті була розроблена інтерактивна система пошуку з використанням поточного місцеположення користувача. В ході виконання робіт було проведено аналіз предметної області інтерактивних систем пошук з використанням поточного місцеположення користувача.

Здійснено дослідження вимог до системи в цілому, вимог до окремих функцій системи, програмного і технічного забезпечення.

Було проведено аналіз та порівняння постачальника географічних даних та на основі цього аналізу обрано постачальника даних на якому базується робота системи. Додатково було проведено аналіз існуючих інтерактивних платформ взаємодії з користувачем. За результатами роботи було побудовано систему з використанням Telegram месенджер як платформу для інтерактивного спілкування з користувачем; Telegram Bot API як спосіб комунікації інтерактивного оточення з серверною частиною; мову програмування Python; Google Maps Platform як постачальника даних; операційну систему Linux для розміщення серверної частини у розподіленому оточенні сервісів DigitalOcean.

Результатом проведеної роботи стала працююча модель інтерактивної системи пошуку інформації з використанням поточного місцеположення користувача.

Подальшими напрямками роботи можуть бути питання що стосуються:

- розробка географічно розподіленого застосунку;
- удосконалення моделі розпізнавання запита від користувача із застосуванням Natural Language Processing;
- розширення функціоналу застосунку.

7. СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дослідження ринку сервісів що використовують поточне місцезнаходження користувача. [Електронний ресурс]. – Режим доступу: <https://www.alliedmarketresearch.com/press-release/location-based-services-market.html> – Назва з екрану.
2. Відомості про використання різних застосунків. [Електронний ресурс]. – Режим доступу: <https://statista.com>
3. Стандарт ISO 7498. [Електронний ресурс]. – Режим доступу: [http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994\(E\).zip](http://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip)
4. Правила використання Telegram Bot API. [Електронний ресурс]. – Режим доступу: <https://core.telegram.org/bots/api>
5. Бібліотека використання Telegram Bot. [Електронний ресурс]. – Режим доступу: <https://github.com/python-telegram-bot/python-telegram-bot>
6. Правила використання Google Maps API [Електронний ресурс]. – Режим доступу: <https://developers.google.com/maps/documentation/>
7. Бібліотека для використання Google Maps API [Електронний ресурс]. – Режим доступу: <https://github.com/googlemaps/google-maps-services-python>
8. Вступ до інтелектуальних підприємств інформатики та мистецтва програмування. [Електронний ресурс]. – Режим доступу: <https://www.edx.org/course/cs50s-introduction-to-computer-science>
9. Проблеми приватності у фейсбук. [Електронний ресурс]. – Режим доступу: <https://www.jurist.org/commentary/2018/04/chris-hoofnagle-facebook-dataism/>
10. Програмування числових методів мовою Python : підруч. / А. В. Анісімов, А. Ю. Дорошенко, С. Д. Погорілий, Я. Ю. Дорогий ; за ред. А. В. Анісімова. – К. : Видавничо-поліграфічний центр "Київський університет", 2014. – 640 с.

11. Використання віртуального оточення у застосунках для мови програмування Python. [Електронний ресурс]. – Режим доступу: <https://virtualenv.pypa.io/en/latest/userguide/>
12. Правила використання бібліотеки geopy. [Електронний ресурс]. – Режим доступу: <https://geopy.readthedocs.io>
13. Конвертація текстових патернів у символи емодзі. . [Електронний ресурс]. – Режим доступу: <https://pypi.org/project/emoji/>
14. Добринін В.Ю. Теория информационно-логических систем / В.Ю. Добринін // Информационный поиск. – Видавництво СпбГУ, 2012. – 38 с.
15. Оліфер В., Оліфер Н., Компьютерні сеті, Видавництво: Пітер, 2016, 992с.
16. Використання системи віртуальних машин DigitalOcean. [Електронний ресурс]. – Режим доступу: <https://www.digitalocean.com/products/droplets/>
17. Адміністрування системи під керівництвом операційної системи Linux. [Електронний ресурс]. – Режим доступу: <https://access.redhat.com/documentation/en/red-hat-enterprise-linux/>
18. Опис функціонування системи розпізнавання та автентифікації користувача FaceID. [Електронний ресурс]. – Режим доступу: <https://9to5mac.com/guides/face-id/>